TAB 26

US005847965A

United States Patent [19]

Cheng

[11] Patent Number: 5,847,965

[45] Date of Patent: Dec. 8, 1998

[54] **METHOD FOR AUTOMATIC ITERATIVE AREA PLACEMENT OF MODULE CELLS IN AN INTEGRATED CIRCUIT LAYOUT**

[75] Inventor: **Chih-liang Eric Cheng**, Milpitas, Calif.

[73] Assignee: **Avant! Corporation**, Fremont, Calif.

[21] Appl. No.: **691,607**

[22] Filed: **Aug. 2, 1996**

[51] **Int. Cl.**[6] **G06F 17/50**

[52] **U.S. Cl.** **364/488**; 364/489; 364/491

[58] **Field of Search** 364/488, 489, 364/490, 491

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,617,714	11/1971	Kernighan et al.	364/488
4,593,363	6/1986	Burstein et al.	364/491
4,908,772	3/1990	Chi	364/491
5,224,056	6/1993	Chene et al.	364/490
5,311,443	5/1994	Crain et al.	364/491
5,495,419	2/1996	Rostoker et al.	364/468
5,513,124	4/1996	Trimberger et al.	364/491
5,557,533	9/1996	Koford et al.	364/491
5,587,923	12/1996	Wang	364/490

OTHER PUBLICATIONS

Breuer, "Min–Cut Placement," Journal of Design Automation & Fault Tolerant Computing, vol. 1, No. 4, Oct. 1977, pp. 343–362.

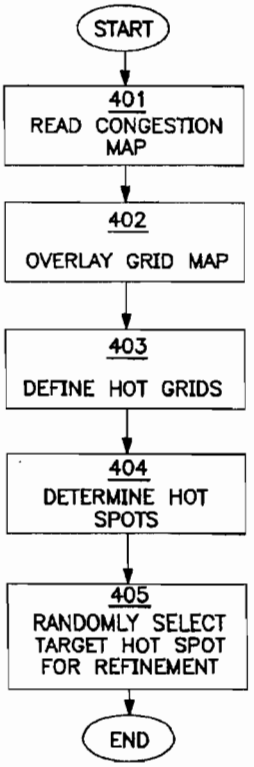
Fiduccia et al., "A Linear–Time Heuristic for Improving Network Partitions," 19th Design Automation Conference, Jun. 1982, pp. 175–181.

Primary Examiner—Kevin J. Teska
Assistant Examiner—Leigh Marie Garbowski
Attorney, Agent, or Firm—Thomas E. Schatzel; Law Offices of Thomas E. Schatzel

[57] **ABSTRACT**

In a computer system, a method for an area based place and route of an integrated circuit layout that provides automatic iterative area placement of module cells intelligently and effectively. In one embodiment, this is accomplished in three phases. The searching phase determines which hot spot is to be refined based on a congestion map. Next, the refining phase chooses a box with the proper aspect ratio, cut line direction, and placement options for minimizing the hot spot. The scheduling phase then decides whether to proceed with another area placement based on the current result or to restore a previous placement that exhibited superior characteristics. In the course of the area placements, several parameters are randomly varied in an intelligent manner so that successive iterative area placements produce equivalent or better results. All of this is accomplished without human intervention or expert knowledge. Instead, the computer system continuously runs its program until a design goal is attained.

8 Claims, 7 Drawing Sheets



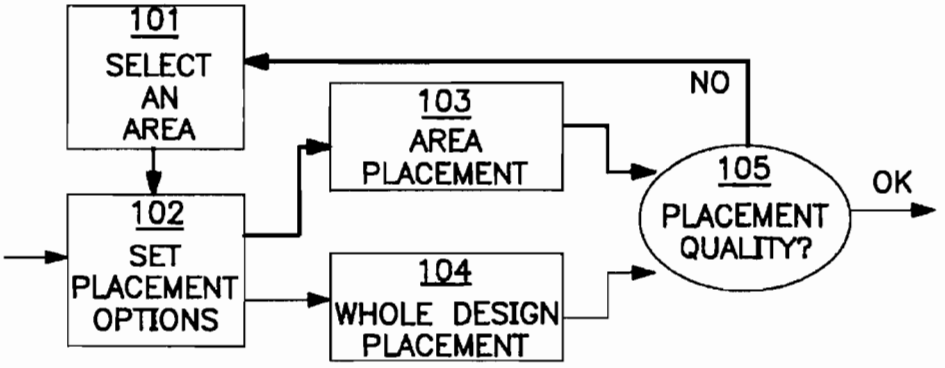


FIG. 1

(PRIOR ART)

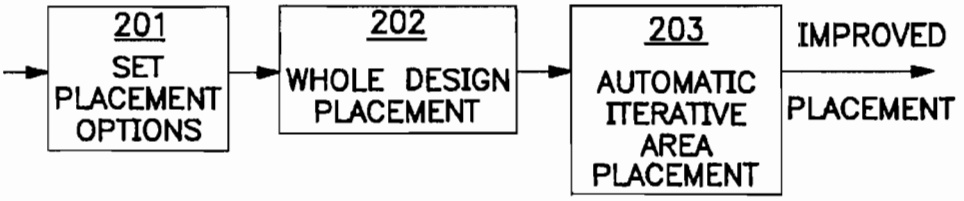


FIG. 2

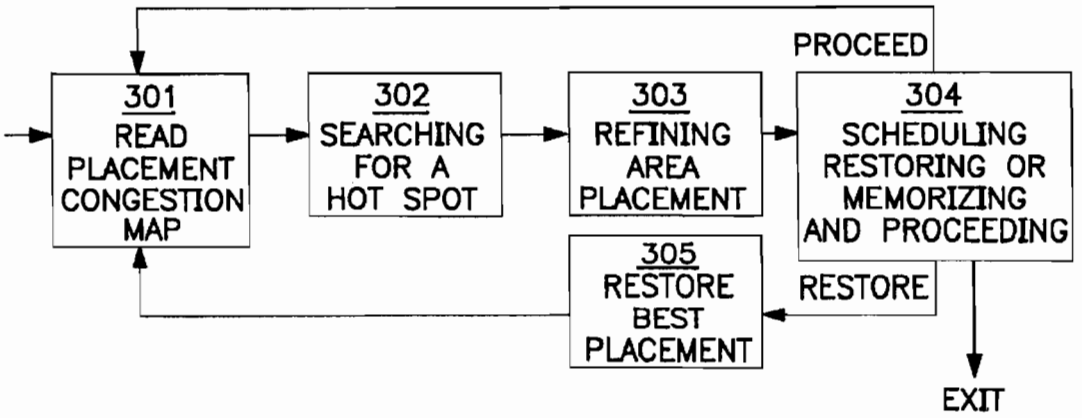


FIG. 3

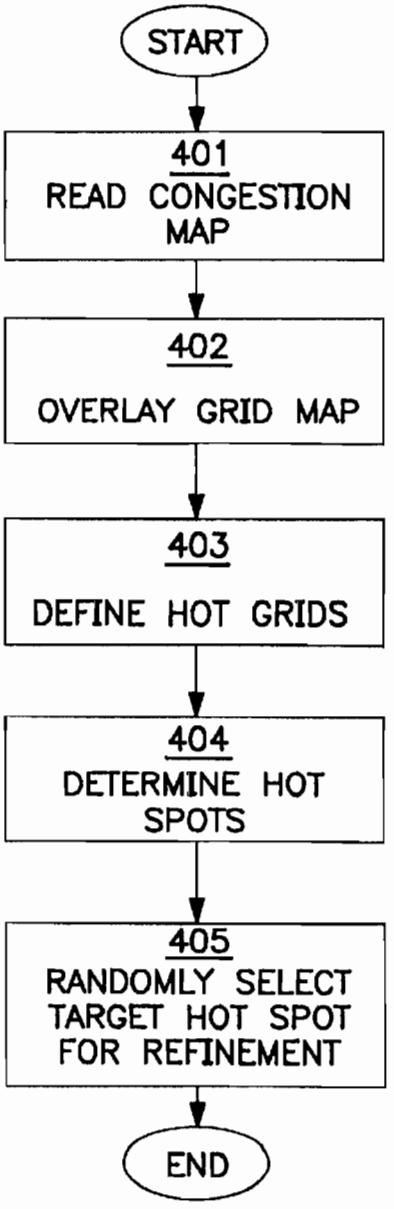

START
401
READ CONGESTION MAP
402
OVERLAY GRID MAP
403
DEFINE HOT GRIDS
404
DETERMINE HOT SPOTS
405
RANDOMLY SELECT TARGET HOT SPOT FOR REFINEMENT
END

FIG. 4

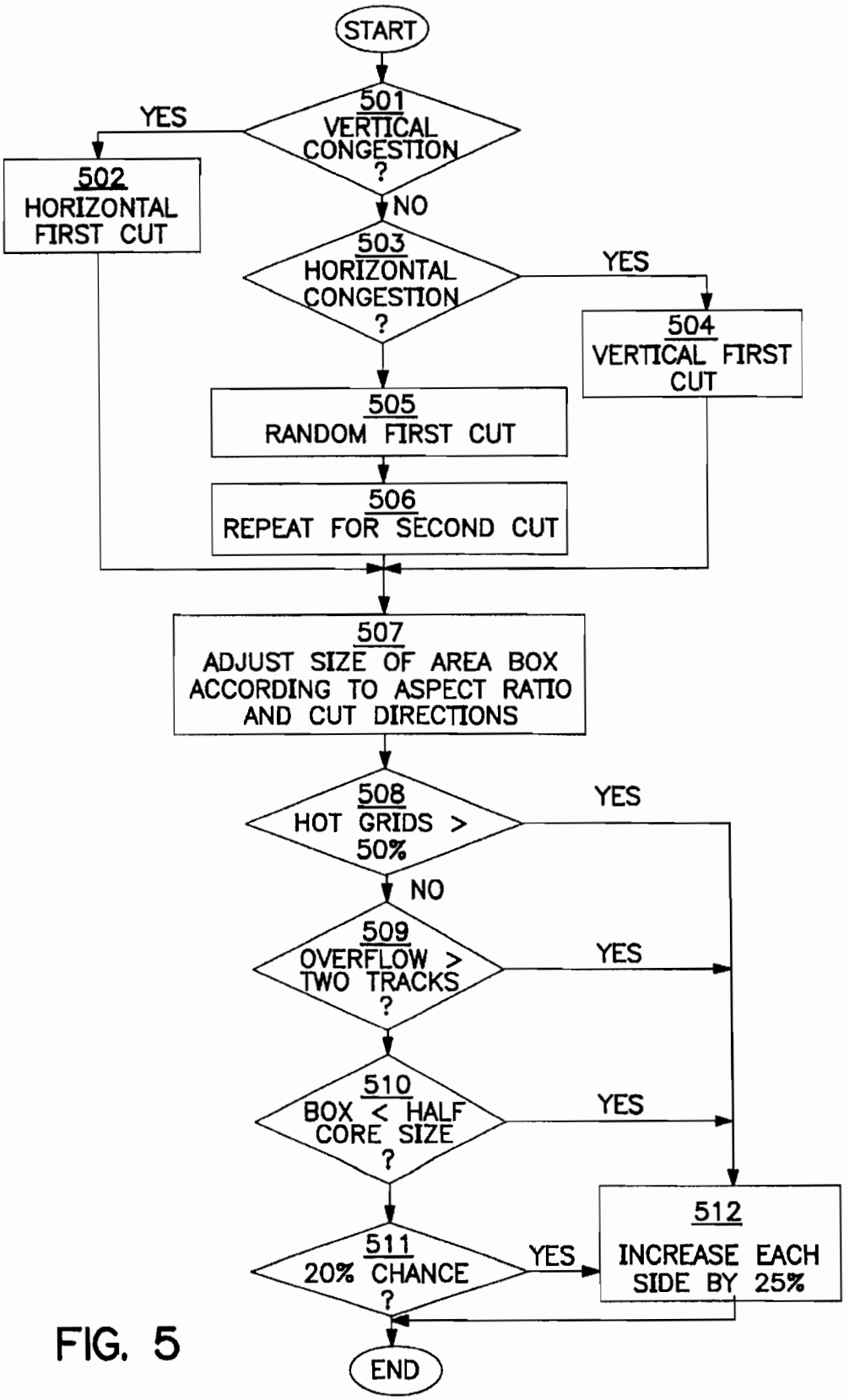

START
501 VERTICAL CONGESTION ?
YES
NO
502 HORIZONTAL FIRST CUT
503 HORIZONTAL CONGESTION ?
YES
504 VERTICAL FIRST CUT
505 RANDOM FIRST CUT
506 REPEAT FOR SECOND CUT
507 ADJUST SIZE OF AREA BOX ACCORDING TO ASPECT RATIO AND CUT DIRECTIONS
508 HOT GRIDS > 50%
YES
NO
509 OVERFLOW > TWO TRACKS ?
YES
510 BOX < HALF CORE SIZE ?
YES
511 20% CHANCE ?
YES
512 INCREASE EACH SIDE BY 25%
END

FIG. 5

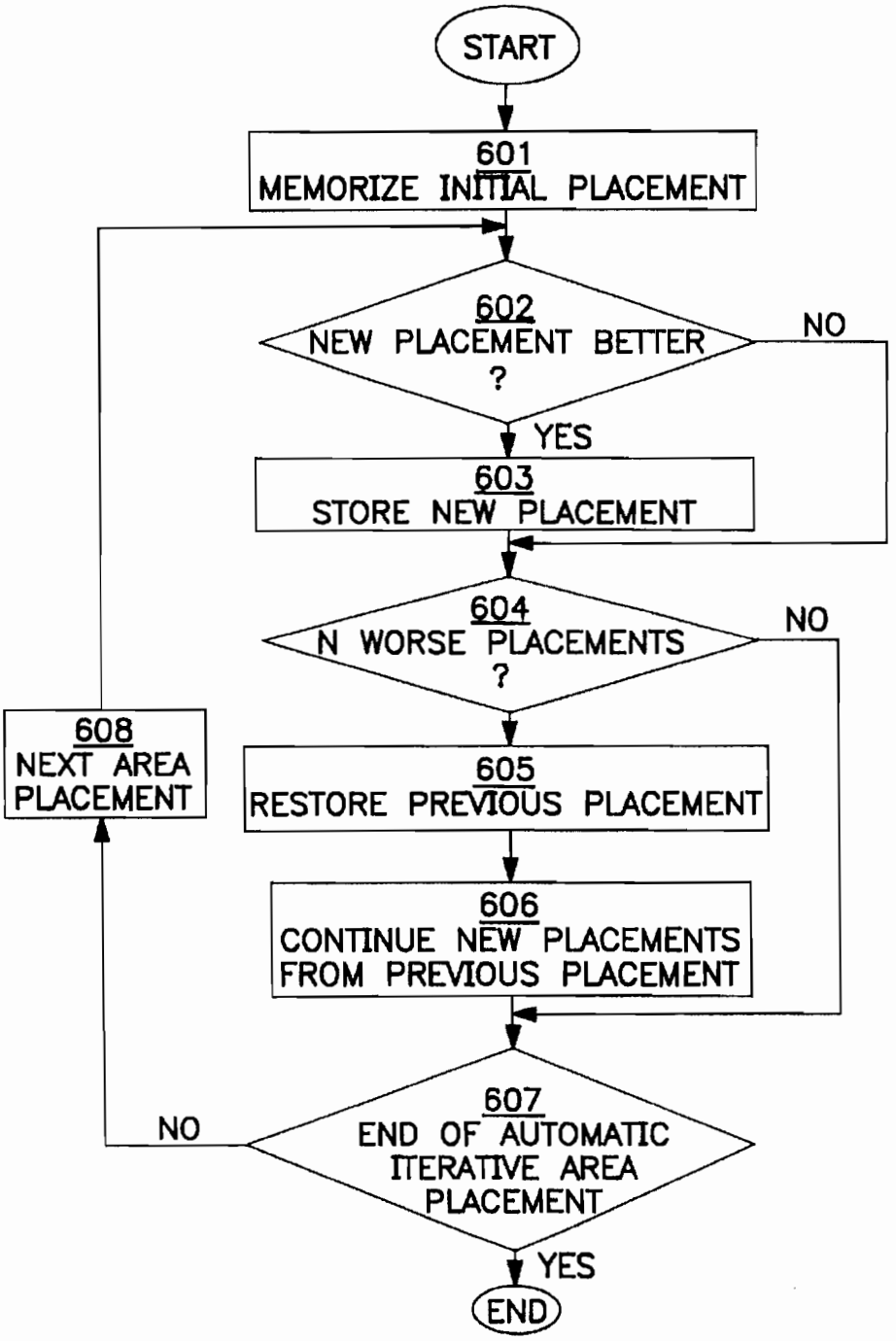

START
601
MEMORIZE INITIAL PLACEMENT
602
NEW PLACEMENT BETTER ?
NO
YES
603
STORE NEW PLACEMENT
604
N WORSE PLACEMENTS ?
NO
608
NEXT AREA PLACEMENT
605
RESTORE PREVIOUS PLACEMENT
606
CONTINUE NEW PLACEMENTS FROM PREVIOUS PLACEMENT
607
END OF AUTOMATIC ITERATIVE AREA PLACEMENT
NO
YES
END

FIG. 6

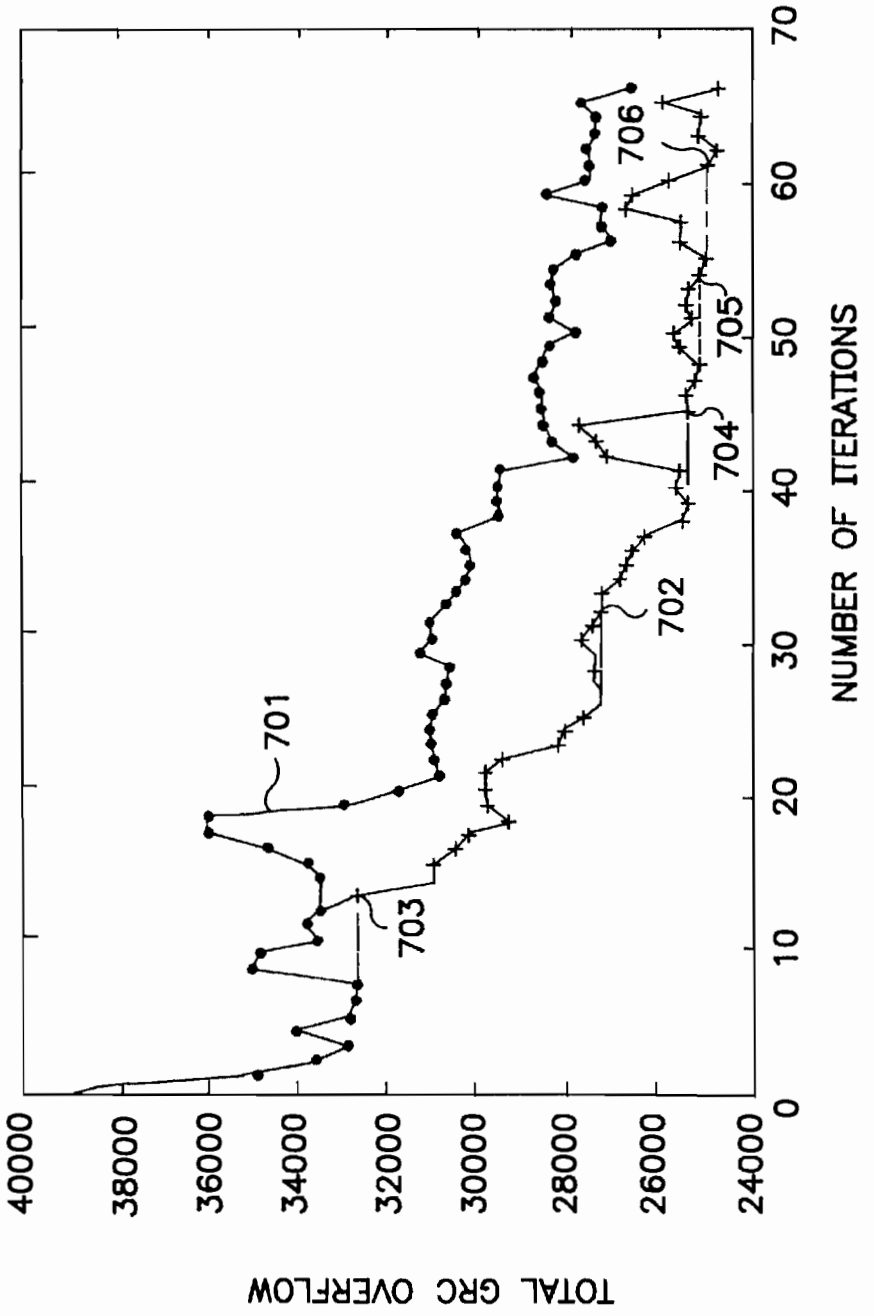

TOTAL GRC OVERFLOW
40000
38000
36000
34000
32000
30000
28000
26000
24000
0
10
20
30
40
50
60
70
NUMBER OF ITERATIONS
701
702
703
704
705
706

FIG. 7

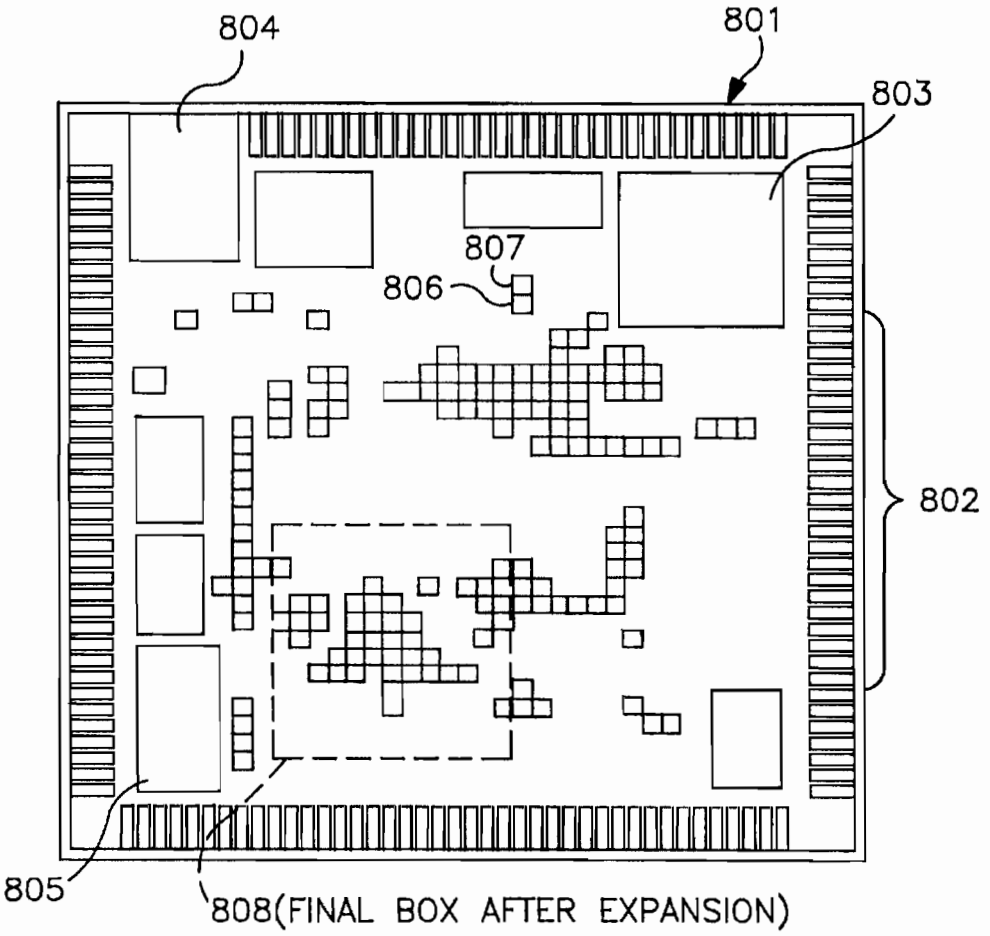


FIG. 8

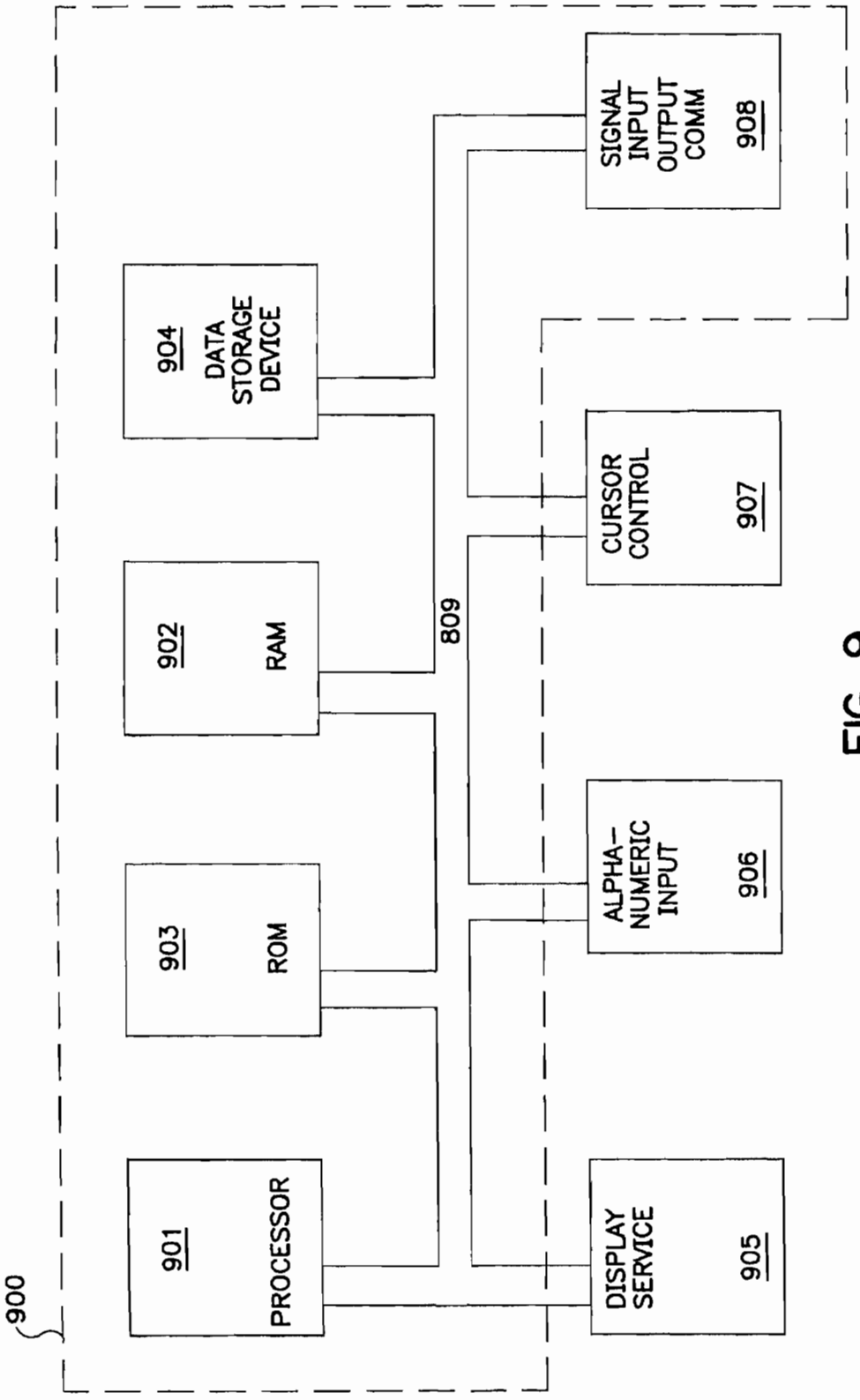

900
901 PROCESSOR
903 ROM
902 RAM
904 DATA STORAGE DEVICE
809
SIGNAL INPUT OUTPUT COMM 908
CURSOR CONTROL 907
ALPHA-NUMERIC INPUT 906
DISPLAY SERVICE 905

FIG. 9

METHOD FOR AUTOMATIC ITERATIVE AREA PLACEMENT OF MODULE CELLS IN AN INTEGRATED CIRCUIT LAYOUT

FIELD OF THE INVENTION

The present invention pertains to a method for the automatic iterative area placement of module cell in a IC semiconductor layout.

BACKGROUND OF THE INVENTION

Advances in semiconductor technology have led the way towards more versatile, powerful, and faster integrated circuit (IC) chips in the fields of computer systems, telecommunications, instrumentation, etc. The trend is towards even larger, more complex and sophisticated IC chips in an effort to meet and improve upon the demands imposed by state-of-the-art performance. Today, a single IC chip can contain upwards of millions of transistors. As the complexity, functionalities, speed, and size of these chips increase, it is becoming a much more critical and difficult task to properly design, layout, and test the next generation of chips.

In order to meet these demands, a highly specialized field, commonly referred to as "electronic design automation" (EDA), has evolved, whereby computers are extensively used to automate the design, layout, and testing process. Indeed, it has now come to the point where the process has become so overwhelming that integrated circuits cannot be designed without the help of computer-aided design (CAD) systems. Computers are ideally suited to these tasks because they can be programmed to reduce or decompose large, complicated circuit designs into a multitude of much simpler functions. Whereupon, the computers can be programmed to iteratively solve these much simpler functions.

Typically, the process begins with an engineer defining the input/output signals, desired functionalities, and performance characteristics of the new IC chip. This information is fed into a logic synthesis program which generates a specification defining the integrated circuit in terms of a particular semiconductor technology (e.g., very large scale integration—VLSI). This specification can be regarded as a template for the realization of the physical embodiment of the integrated circuit in terms of transistors, routing resources, etc. Next, a place and route CAD tool is used to determine the routing, pinouts, wiring, interconnections and general physical layout of the chip.

One common method used in the place and route of a chip involves a grid-based channel approach. In this approach, the routing area (i.e., the channel) of the chip is adjusted according to the demands imposed by the functionality and routing requirements. In other words, the channel is widened if additional routing resources are required. And conversely, the channel can be shrunk to eliminate excess routing capacity. The size of the layout is determined only at the completion of the overall process. Of course, the disadvantage to this grid-based channel process is that it is extremely difficult for the designer to control the size or area of the silicon die. The human designer has minimal control over the die size and must accept whatever is generated by the channel router.

An alternative method for the place and route of a semiconductor chip is that of an area-based approach. In an area-based approach, a specific area of silicon is prespecified. The software then attempts to layout the design within this given size constraint. Often, quadratic programming is applied to place the circuit of all module cells. This

approach works fairly well for simple layouts. However, problems may be encountered with larger, more complex layouts requiring denser circuitry. The problem is that there may be certain sections which require higher amounts of routing resources. For example, data path circuitry require more routing resources versus decoder circuitry, which tend to be more locally connected. As a result, layouts tend to have sections with high concentrations of overlapping interconnections. These "clusters" or "hot spots" are highly undesirable. Sometimes, the congestion within a hotspot is so severe that it becomes physically impossible to complete the routing of nets. And even if it were possible to route all of the nets, it might require an extraordinary amount of processing power, time, and skill in order to finally achieve the objective.

It is highly desirable to more evenly distribute or spread out the routing so as to minimize hot spots. Consequently, many state-of-the-art layout systems provide an "area placement" option which allows designers to manually select particular areas and set placement options to refine the placement of module cells within those selected areas. Thereby, designers can manually intervene in the layout process in an attempt to minimize the hotspot areas. FIG. **1** shows a usage flow of a typical prior art placer. In step **101**, the designer selects a particular area of interest. Next, the designer judiciously sets the placement options, step **102**. Based thereon, the software program performs area placement, step **103**, and also a whole design placement, step **104**. Afterwards, the designer examines the results and decides whether it is acceptable. If the result does not meet the design criteria, the process must be repeated. It is not uncommon for a small, complex IC design to take approximately ten to twenty separate area placements with placement options properly set before the module cell placement becomes refined to a routable level.

Clearly, this manual area placement option is extremely labor-intensive, tedious, and time-consuming. Furthermore, it requires a highly skilled designer with specialized knowledge and a vast amount of practical experience. It takes intensive training to become an expert in performing area placement correctly, efficiently and effectively. First, the designer must be able to identify hot spots from the placement of cells. The designer must then find the box containing a hot spot within which module cells are to be placed. Next, the designer must properly set the placement congestion removal option to perform area placement. Finally, the designer must select cut line directions for the placement to proceed. This process is repeated over and over again many times. And each time that the process is performed, it can take the computer many minutes or even hours to execute. In the meantime, this highly paid designer is idle and must wait for the outcome.

Therefore, there is a great need in the prior art for some mechanism that could somehow relieve the burden of area placement from human designers. The present invention provides a solution to this problem that is inexpensive and yet highly effective. The present invention accomplishes this feat by applying a computer-implemented process which performs automatic iterative area placement of module cells in an IC layout without human interaction.

SUMMARY OF THE INVENTION

The present invention pertains to a computer implemented method for an area based place and route of an integrated circuit layout that provides automatic iterative area placement of module cells. Rather than having a skilled human

having expert knowledge performing the task of area placement, the present invention accomplishes the same function automatically under software guidance. There are a number of different ways in which the present invention of an automatic area placement can be implemented. In one embodiment, this is accomplished in three phases. The searching phase determines which hot spot is to be refined based on a congestion map. The congestion map contains routing information. A grid is then overlaid on top of the congestion map. Those grids having a high degree of congestion is designated as being hot grids. Groups of abutting hot grids are designated as being hot spots. A hot spots is then selected for refinement. The selection is randomly based on intelligently varying certain parameters so that successive area placements produce different results. In the refining phase, a box containing the selected hot spot is chosen. The computer system automatically sets the proper aspect ratio, cut line direction, and placement options for minimizing the hot spot. Finally, the scheduling phase decides whether to proceed with another area placement based on the current result or to restore a previous placement that exhibited superior characteristics so that successive iterative area placements produce equivalent or better results. In the present invention, all of this is accomplished without human intervention. Instead, the computer system continuously runs its program until a design goal is attained.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

FIG. **1** shows a usage flow of a typical prior art placer.

FIG. **2** shows a computer-implemented placement usage flow according to the present invention.

FIG. **3** shows a more detailed flowchart describing the steps for performing the automatic iterative area placement flow according to the currently preferred embodiment of the present invention.

FIG. **4** is a flowchart describing the steps for the searching phase.

FIG. **5** is a flowchart describing the steps for the refining phase.

FIG. **6** is a flowchart describing the steps for the scheduling phase.

FIG. **7** shows a sample chart of the total GRC overflow as a function of the number of iterations performed.

FIG. **8** shows an example of a congestion map.

FIG. **9** illustrates an exemplary computer system upon which the present invention may be implemented or practiced.

DETAILED DESCRIPTION

An automatic iterative area placement of module cells in an IC layout is described. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be obvious, however, to one skilled in the art that the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to avoid obscuring the present invention. It should be noted that the present invention is capable of performing whole design placement of module cells or placement within a given area.

FIG. **2** shows a computer-implemented placement usage flow according to the present invention. In step **201**, the placement options are set. Next, the whole design placement is constructed, step **202**. Finally, an automatic iterative area placement is performed, step **203**. Note that there is no need for human intervention, nor is there a requirement for the process to be manually repeated. Instead, the computer system automatically finds the most optimal area placement. When finished, the computer system notifies the human designer of the completed layout. Thus, there is no need for the designer to have any expert knowledge in identifying hot spots, defining boxes, setting congestion options, etc. With the present invention, the designer simply inputs the initial conditions and hits a button to start the process. From thereon, the computer system automatically performs the area placement. Thus, with the present invention, any general user can confidently, reliably, and effectively perform area placement without having to know area selection or placement option setting.

The currently preferred embodiment of the present invention consists of the following three phases: searching, refining, and scheduling. Basically, the searching phase determines which hot spot is to be refined. This determination is based on a congestion map created by the placement engine. The refining phase then chooses a box with the proper aspect ratio, cut line directions, and placement options that is appropriate for the selected area to be refined. Lastly, the scheduling phase decides whether the area placement should be repeated again. This determination is based on comparing the current result against the best prior placement which was memorized during the refining phase.

FIG. **3** shows a more detailed flowchart describing the steps for performing the automatic iterative area placement flow according to the currently preferred embodiment of the present invention. Initially, the computer system reads the placement congestion map, step **301**. The congestion map is generated by the placement engine. Based on the congestion map, the computer system searches for a hot spot, step **302**. Once a hot spot has been determined, the hot spot and its surrounding area is then refined, step **303**. After refinement has been completed, the scheduling phase is performed, step **304**. In the scheduling phase, a decision is made as to whether it is better to proceed with further area placement or to restore a previous placement. If the decision is to proceed, then the process is repeated at step **301**. Otherwise, the best prior placement is restored, step **305**, and the process repeats again at step **301**. After a predetermined number of iterations or a runtime limit until a desired goal has been attained, the process exits. Each of the searching, refining, and scheduling phases are now described in detail below.

FIG. **4** is a flowchart describing the steps for the searching phase. First, the searching phase starts with the reading of a congestion map, step **401**. The congestion map is a two-dimensional display of the routing for a particular placement. The core area is partitioned into a two-dimensional array of grids. The grids, also known as buckets or bins, contain an N×M number of routing tracks (e.g., wires), extending in both horizontal and vertical directions. The size of the grids is randomly set within a design-dependent range. This randomness of the grid size is purposely made to vary the cut line location. The grid map is then overlapped on top of the congestion map, step **402**. If a grid contains congestion judged by a chosen threshold, the grid is defined to be a "hot" grid, step **403**. This threshold is also randomly varied within a preset range in order to provide a degree of variation of the congestion resolution. Once the hot grids have been defined, a hot spot is found in step **404** by

grouping a number of hot grids that abut one another either on top, bottom, left, or right edge boundaries. This can be achieved by visiting neighboring hot grids recursively while preventing revisiting.

After all hot spots have been identified, a particular hot spot is randomly selected as the target hot spot to resolve within the current iteration, step **405**. The probability of selection of a hot spot is weighted according to its particular size. Larger hot spots have a higher probability of being selected as a target hot spot. However, it is important to give smaller hot spots a chance of being selected so that they can also be improved upon. In the currently preferred embodiment, all M hot spots are sorted based upon their area and a probability of P is imposed on the number N hot spot, in which P is given as:

$$P = \frac{1/N}{\sum_{n=1}^{M} 1/n}$$

For example, if M=5, then the probability of getting selected as the target hot spot is 44%, 22%, 14%, 11%, and 9% for each hot spot as sorted by their respective sizes.

FIG. **5** is a flowchart describing the steps for the refining phase. Basically, the refining phase performs two functions: (1) determining first and second cut line directions and (2) finding a box to contain the target hot spot. The cut line direction is determined as follows. It is known that the congestion map contains both vertical and horizontal routing congestion information. For vertical congestion, the placement engine calls for a horizontal first cut to reduce vertical congestion, steps **501–502**, and vice versa for horizontal congestion, steps **503–504**. By analyzing the target hot spot, the computer system can determine which direction is more congested. Assume that V and H represent the congestion cost (e.g., the number of tracks) of vertical and horizontal congestion. If (V>1.2*H), then the first cut is chosen to be horizontal. Likewise, if (H>1.2*V), then the first cut is chosen to be vertical. Otherwise, the first cut direction is randomly selected between vertical and horizontal directions, step **505**. Furthermore, if (V>2*H), then the second cut is also horizontal, and if (H>2*V), then the second cut is also vertical, step **506**.

The area box is determined as follows. Basically, the bounding box of the hot spot is found to be the initial area box. The following adjustments to the initial area box are done to optimize the selection, step **507**. Assume that R represents the aspect ratio of the area box, where R=Y/X, with Y being the height of the box and X being the width of the box. If the first cut is horizontal and R<1, then Y is increased to make R=1. If the first cut is vertical and R>1, then X is increased to make R=1. If the second cut is also horizontal, same as the first cut, Y is further increased so that R=1.5. Likewise, if the second cut is also vertical, same as the first cut, X is further increased so that R=0.67. The area box is then extended by one grid on the top, bottom, left, and right boundaries. If one of the following four conditions is true, the area box is expanded by 25% on each side according to step **512**:

1) Hot grids comprise more than 50% of all grids in the current area box (step **508**);

2) Average overflow of more than two tracks for each global routing cell (GRC) in a grid (step **509**);

3) The current area box is smaller than half of the core box in both dimensions (step **510**); and

4) Twenty percent random chance to expand the box (step **511**).

The area box after adjustment is centered at the target hot spot. If the selected area box is not fully inside the core area box, the area box is shifted such that it is fully inside the core area box.

In addition to the first and second cut directions, which are placement options, the computer system also switches between two congestion-driven options of the placement engine: "padding" and "spacing." There are situations where the spacing option is used rather than the padding option. First, spacing is used if there are region constraints in the design. Second, spacing is used if the selected box overlaps with no less than one macro block. And third, spacing is used in 50% of chance by random probability. Otherwise, the padding option is used.

The scheduling phase is now described in detail with reference to the flowchart of FIG. **6**. The scheduling phase is performed because of the fact that not all area placement results in a better placement than that which has existed before. Thus, the scheduling phase takes the following steps to memorize the best placement. The term "best" is defined by a number called "total GRC overflow" when a congestion map is updated or created. First, the initial placement is memorized, step **601**. A comparison is made between the new placement and the prior best placement, step **602**. If the next area placement iteration results in a better placement, that new placement is memorized (e.g., stored into computer memory) as being the best, step **603**. If it is determined in step **604** that for consecutive N area placements, all of which result in worse placement than the previously memorized best placement, the memorized best placement is restored, step **605**. Thereupon, the area placement continues with its processing according to the newly restored best placement, step **606**. This process is repeated until the end of the automatic iterative area placement is achieved, steps **607–608**.

It should be noted that the restoration of the placement does not result in repetition of history because a random number generator is widely used in the present invention in both the searching and refining phases. Hence, a different hot spot, a different box size, and/or different placement options may be used for subsequent placements which differ from those of previous placements. Thereby, additional placements will gradually produce better and better results rather than worse results. In this manner, the memorizing and restoration of the best placement enables the drastic improvement on placement quality, which is based on the total GRC overflow. Because of the scheduling phase, users always get an equivalent (in a worst case scenario) or a much better placement (in most other cases) at the end of the automatic iterative area placement.

FIG. **7** shows a sample chart of the total GRC overflow as a function of the number of iterations performed. The top plot **701** shows the case where scheduling is not applied. Whereas, the bottom plot **702** has the benefit of the scheduling enhancement. It can be seen that there are peaks and valleys. Sometimes successive iterations reduce the total GRC overflow, whereas other times the total GRC overflow is actually increased or worsened. In this example, there are four instances **703–706** where a successive area placement is worse than a previous area placement. In these instances, a rollback operation is performed to rollback the area placement so that it is better.

In the currently preferred embodiment, there are four criteria to stop the searching and refining phases:

1) Set total number of areas to search and refine;

2) Set total run time limit;

3) Set maximum overflowing GRC number; and

4) Set overflowing GRC percentage.

If either one of these four holds true, the process stops and the best placement memorized is restored.

FIG. **8** shows a typical congestion map **801**. The congestion map **801** contains a number of peripheral input/output (I/O) cells **802** and some macrocells **803–805**. Other circuitry is placed in a host of module cells within the remaining silicon area. Certain areas which have high degree of congestion are represented by small squares, known as "hot grids." For example, a small area of congestion is represented by the two hot grids **806** and **807**. Based on the hot grids, the computer system selects a box to encompass the surrounding regions of a particular hot spot. An example of such a bounding box is shown as **808**. Area placement is performed within the bounding box **808**. Once the box is determined, the computer system must decide whether to make the first cut in a horizontal or vertical direction. In this example, a horizontal cut is shown. Refinement is then performed on this box **808**.

FIG. **9** illustrates an exemplary computer system **900** upon which the present invention may be implemented or practiced. It is appreciated that the computer system **900** of FIG. **9** is exemplary only and that the present invention can operate within a number of different computer systems including general purpose computers systems, embedded computer systems, and computer systems specially adapted for electronic design automation. Computer system **900** of FIG. **9** includes an address/data bus **909** for conveying digital information between the various components, a central processor unit (CPU) **901** for processing the digital information and instructions, a random access memory (RAM) **902** for storing the digital information and instructions, a read only memory (ROM) **903** for storing information and instructions of a more permanent nature. In addition, computer system **900** may also include a data storage device **804** (e.g., a magnetic, optical, floppy, or tape drive) for storing vast amounts of data, and an I/O interface **908** for interfacing with peripheral devices (e.g., computer network, modem, etc.). Devices which may be coupled to computer system **900** include a display device **905** for displaying information (e.g., channel grid map, vertical constraint graphs, weighting functions, feasible links, etc.) to a computer user, an alphanumeric input device **906** (e.g., a keyboard), and a cursor control device **907** (e.g., mouse, trackball, light pen, etc.) for inputting data and selections.

Thus, an automatic iterative area placement of module cells in an IC layout is described. The present invention successfully replaces the knowledge of an expert who sits in front of a workstation and does box selection, option setting repetitively one area after another. The present invention greatly facilitates the useability of the tools for general users in achieving the same or even better results that an expert could get. Moreover, the present invention saves the expert precious time from sitting in front of the workstation or waiting for the results of each successive iteration. In short, the present invention offers a huge improvement in useability of tools for all users and in minimizing congestion for the layout of IC's.

The foregoing descriptions of specific embodiments of the present invention have been presented for purposes of illustration and description. They are not intended to be exhaustive or to limit the invention to the precise forms disclosed, and obviously many modifications and variations are possible in light of the above teaching. The embodiments were chosen and described in order to best explain the principles of the invention and its practical application, to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the Claims appended hereto and their equivalents.

What is claimed is:

1. A computer-implemented automation tool for automatic area placement in designing circuit layouts for semiconductor integrated circuits, comprising:

a computer platform for processing a plurality of program execution modules which accept a set of placement options and initial conditions related to a semiconductor integrated circuit design, computes an area placement solution, and that outputs a completed layout that is used as a guide for the fabrication of a semiconductor integrated circuit;

a first computer process for searching in an area-placement-solution computation for a hot-spot candidate with a relatively high concentration of overlapping connections for further refining, and for loading and execution on the computer platform as one of said program execution modules;

a second computer process for refining said hot-spot candidate by matching it with an area box having a particular combination of aspect ratio, cut-line direction, and placement options, and for loading and execution on the computer platform as one of said program execution modules; and

a third computer process connected to receive a plurality of refined ones of said hot-spot candidates, for scheduling area-placement-solution-computation repetitions, and for loading and execution on the computer platform as one of said program execution modules.

2. A computer-implemented automation tool for automatic area placement in designing circuit layouts for semiconductor integrated circuits, comprising:

a placement engine that calls for horizontal cuts to reduce vertical congestion and vertical cuts to reduce horizontal congestion, and providing for a generating of a plurality of placement congestion maps for a circuit layout for a semiconductor integrated circuit, wherein each of said placement congestion maps is a two-dimensional routing display for a particular placement and contains both vertical and horizontal routing congestion information;

a search mechanism providing for an identification within said plurality of placement congestion maps of a target hot-spot candidate with a relatively high concentration of overlapping connections;

a refining mechanism providing for a refined target hot-spot from said identified target hot-spot candidate and a surrounding adjacent area; and

a scheduling mechanism providing for a decision whether to proceed with a placement solution that includes said refined target hot-spot candidate and then repeat, or to restore a previous placement solution and then repeat, and wherein said decision is based on at least one of a count of a number of iterations, a run-time limit, and an attained placement goal.

3. The automation tool of claim **2**, wherein:

the search mechanism further provides for partitioning a core area into a two-dimensional array of grids that are randomly sized within a design-dependent range to vary a cut line location, and that each contain an N×M array of routing tracks; and

the search mechanism also provides for a grid map comprising said two-dimensional array of grids that is

overlapped on top of at least one of said plurality of placement congestion maps to judge if any grid includes a hot-grid that is defined to be one that exceeds a congestion-threshold variable.

4. The automation tool of claim **3**, wherein:

the search mechanism further provides for finding a plurality of hot-spots by grouping for each a plurality of said hot-grids that abut one another on a boundary edge.

5. The automation tool of claim **2**, wherein:

the refining mechanism further includes determining a first and a second cut-line direction, and finding an area box to contain said target hot-spot;

wherein, said first and a second cut-line directions are each determined by analyzing a target hot-spot to determine a more-congested direction; and

wherein, said area box is determined by starting with initial area box equal to a bounding box for a hot spot, and

adjusting for a first cut at least one of a height (X) and a width (Y) dimension of said initial area box to achieve a first particular aspect ratio of said width and a height dimensions (R=Y/X), and

if a first and a second cut are in a same direction, adjusting for at least one of a height (X) and a width (Y) dimension of said initial area box to achieve a second particular aspect ratio of said width and a height dimensions (R=Y/X).

6. The automation tool of claim **5**, wherein:

the refining mechanism further includes additional expansion of said initial area box if;

more than half of all grids in a current area box are hot grids;

there is an average overflow of more than two tracks for each global routing cell in a grid;

a current area box is smaller than half of said core box in two dimensions; or

there is at least twenty percent random chance to expand said initial area box;

wherein, an objective of an area box adjustment is centered at the target hot spot. If the selected area box is not fully inside the core area box, the area box is shifted such that it is fully inside the core area box.

7. The automation tool of claim **2**, wherein:

the searching mechanism further provides for a comparison between a current placement and at least one previous placement, and that then provides for a retention of one of said current and previous placements as being a best placement depending on a total global-routing-cell overflow value associated with each placement when a congestion map is processed.

8. A computer-implemented method for automatic area placement in designing circuit layouts for semiconductor integrated circuits, comprising the steps of:

computer processing a plurality of program execution modules which accept a set of placement options and initial conditions related to a semiconductor integrated circuit design, and that then compute an area placement solution, and output a completed layout for use as a fabrication guide for a semiconductor integrated circuit;

using one of one of said program execution modules to search in an area-placement-solution computation for a hot-spot candidate with a relatively high concentration of overlapping connections for further refining;

using one of one of said program execution modules to refine said hot-spot candidate by matching it with an area box having a particular combination of aspect ratio, cut-line direction, and placement options; and

using one of one of said program execution modules to receive a plurality of refined ones of said hot-spot candidates and schedule area-placement-solution-computation repetitions.

* * * * *

TAB 27

US006442743B1

(12) **United States Patent**
Sarrafzadeh et al.

(10) **Patent No.: US 6,442,743 B1**
(45) **Date of Patent: Aug. 27, 2002**

(54) **PLACEMENT METHOD FOR INTEGRATED CIRCUIT DESIGN USING TOPO-CLUSTERING**

(75) Inventors: **Majid Sarrafzadeh**, Wilamette, IL (US); **Lawrence Pileggi**, Pittsburgh, PA (US); **Sharad Malik**, Princeton, NJ (US); **Feroze Peshotan Taraporevala**, San Jose, CA (US); **Abhijeet Chakraborty**, Sunnyvale, CA (US); **Gary K. Yeap**, San Jose, CA (US); **Salil R. Raje**, Santa Clara, CA (US); **Lilly Shieh**, Union City, CA (US); **Douglas B. Boyle**, Palo Alto, CA (US); **Dennis Yamamoto**, Los Altos, CA (US)

(73) Assignee: **Monterey Design Systems**, Sunnyvale, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/097,107**

(22) Filed: **Jun. 12, 1998**

(51) **Int. Cl.**[7] **G06F 17/50**
(52) **U.S. Cl.** **716/10**; 716/9
(58) **Field of Search** 716/10, 9, 8, 7, 716/12, 13, 14

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,491,641	A		2/1996	Scepanovic et al. 716/13
5,495,419	A		2/1996	Rostoker et al. 700/121
5,526,517	A		6/1996	Jones et al. 707/8
5,557,533	A		9/1996	Koford et al. 716/9
5,636,125	A		6/1997	Rostoker et al. 700/121
5,661,663	A	*	8/1997	Scepanovic et al. 364/490
5,682,321	A		10/1997	Ding et al. 716/8
5,682,322	A		10/1997	Boyle et al. 716/2
5,699,265	A		12/1997	Scepanovic et al. 716/10
5,712,793	A	*	1/1998	Scepanovic et al. 364/490
5,748,844	A	*	5/1998	Marks 706/45
5,838,585	A	*	11/1998	Scepanovic et al. 364/491
5,854,752	A	*	12/1998	Agarwal 364/489
5,903,461	A	*	5/1999	Rostoker et al. 364/468.28
5,909,376	A	*	6/1999	Scepanovic et al. 716/8
5,955,776	A	*	9/1999	Ishikawa 257/618
6,030,110	A	*	2/2000	Scepanovic et al. 364/490
6,067,409	A	*	5/2000	Scepanovic et al. ... 395/500.09

OTHER PUBLICATIONS

Fiduccia et al., "A Linear–Time Heuristic for Improving Network Partitions", *IEEE 19th Design Automation Conference*, pp. 175–181 (1982).

Huang et al., "Partitioning–Based Standard–Cell Global Placement With an Exact Objective", *International Symposium on Physical Design*, pp. 18–25, ACM (1997).

Kernighan et al., "An Efficient Heuristic Procedure for Partitioning Graphs", *Bell System Technical Journal*, 49:291–307 (1970).

Kleinhans et al., "Gordian: VLSI Placement by Quadratic Programming and Slicing Optimization", *IEEE Transactions on Computer–Aided Design*, 10(3):356–365 (1991).

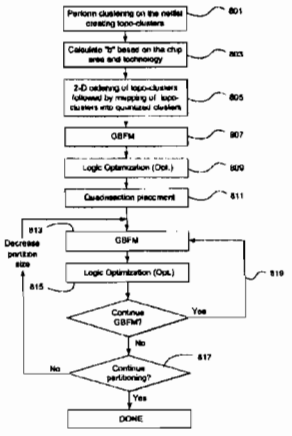
(List continued on next page.)

Primary Examiner—Vuthe Siek
(74) *Attorney, Agent, or Firm*—Vierra Magen Marcus; Harmon & DeNiro LLP

(57) **ABSTRACT**

The disclosure describes a placement method for the physical design of integrated circuits in which natural topological feature clusters are discovered and exploited during the placement process is disclosed. Topo-clusters drive initial placement, with all of the gates of a topo-cluster being placed initially in a single bin of the placement layout or within a group of positionally-related bins. An iterative placement refinement process is done using a technique referred to as Dual Geometrically-Bounded FM (GBFM). GBFM is applied on a local basis to windows encompassing a number of bins. From iteration to iteration, windows may shift position and vary in size. When a region bounded by a window meets a specified cost threshold in terms of a specified cost function, that region stops participating. Following the foregoing global placement process the circuit is then ready for detailed placement in which cells are assigned to placement rows.

13 Claims, 9 Drawing Sheets



OTHER PUBLICATIONS

Sarrafzadeh et al., "NRG: Global and Detailed Placement", *IEEE International Conference on Computer–Aided Design*, pp. 532–537 (1997).

Suaris et al., "Quadrisection: A New Approach to Standard Cell Layout", *IEEE International Conference on Computer–Aided Design*, pp. 474–477 (1987).

* cited by examiner

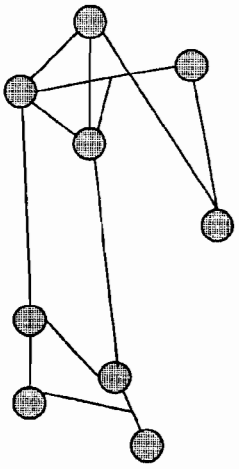


FIG. 1A

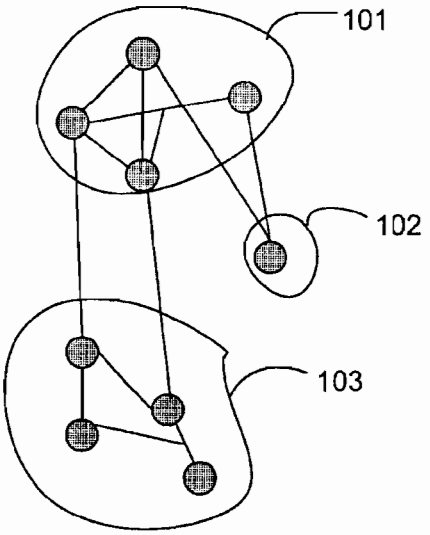


FIG. 1B

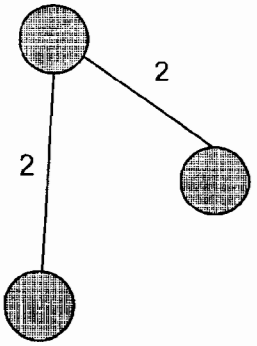


FIG. 1C

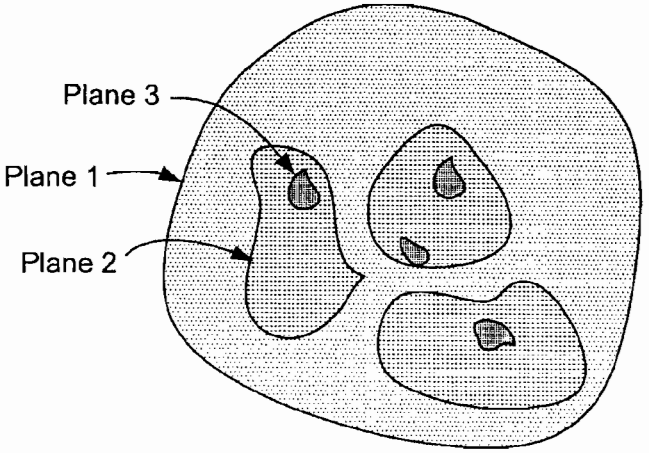


FIG. 2

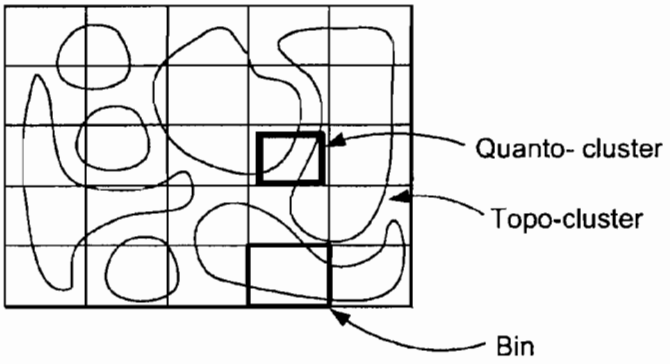


FIG. 3

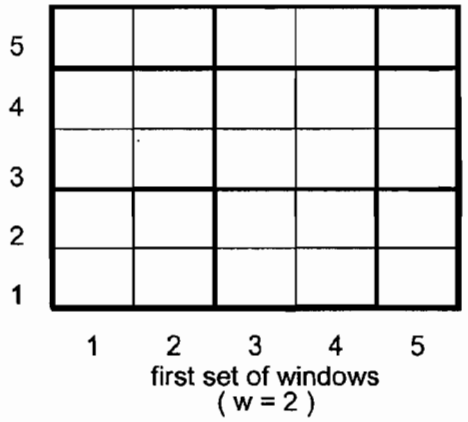


FIG. 4A

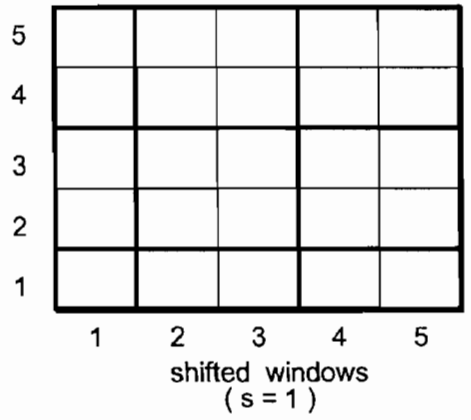


FIG. 4B

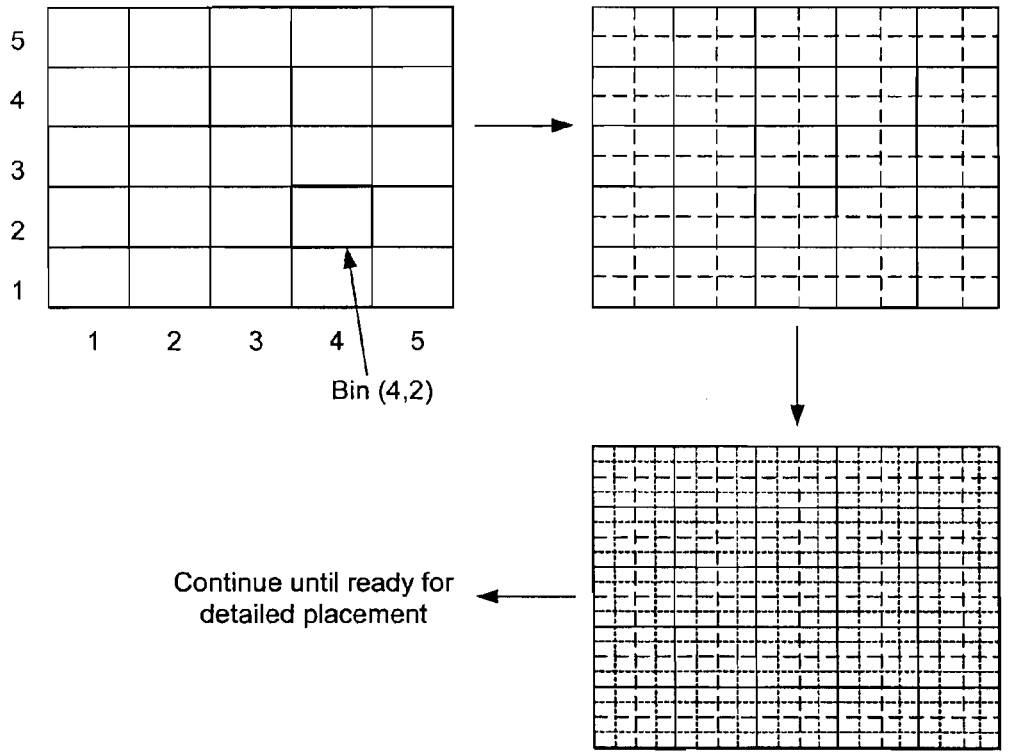

5
4
3
2
1
1 2 3 4 5
Bin (4,2)
Continue until ready for detailed placement

FIG. 5

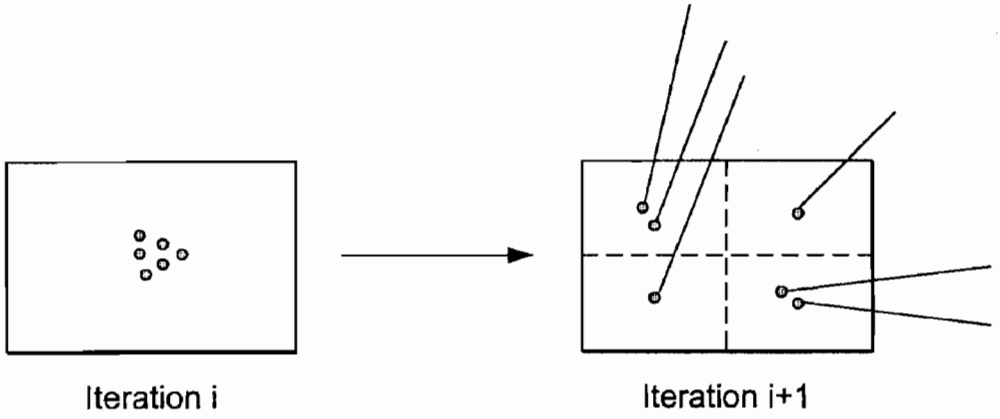

Iteration i
Iteration i+1

FIG. 6

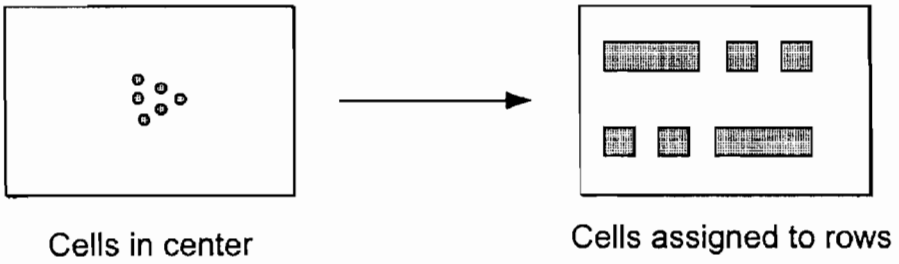

Cells in center
Cells assigned to rows

FIG. 7

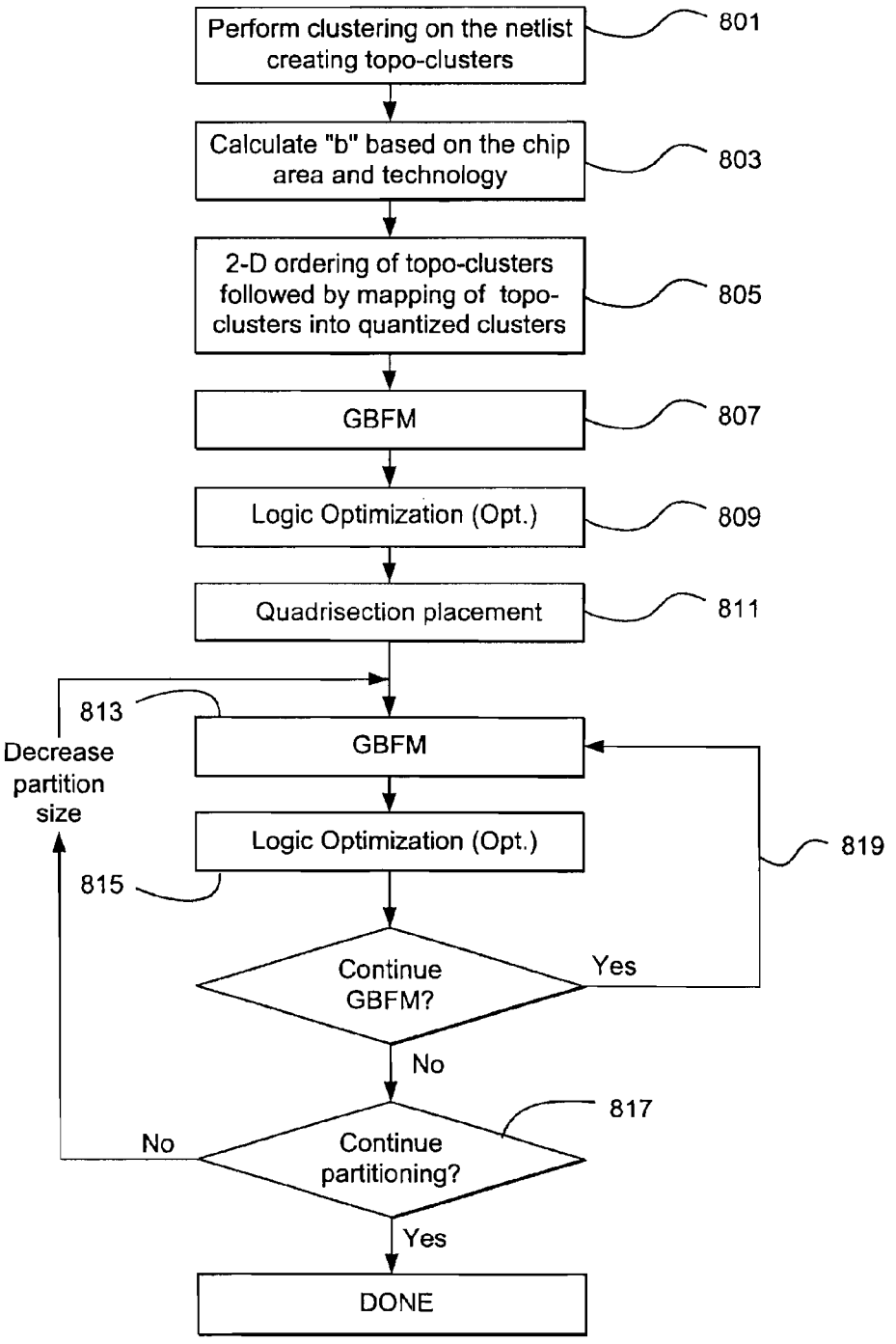

Perform clustering on the netlist creating topo-clusters
801
Calculate "b" based on the chip area and technology
803
2-D ordering of topo-clusters followed by mapping of topo-clusters into quantized clusters
805
GBFM
807
Logic Optimization (Opt.)
809
Quadrisection placement
811
813
Decrease partition size
GBFM
Logic Optimization (Opt.)
815
819
Continue GBFM?
Yes
No
817
No
Continue partitioning?
Yes
DONE

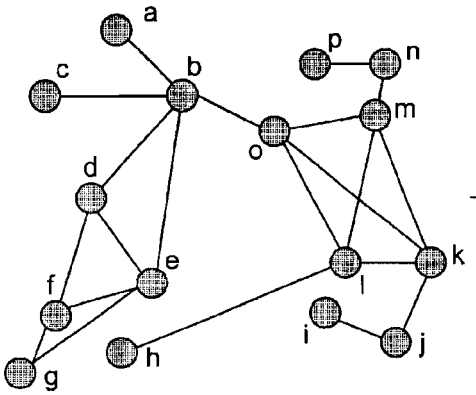
FIG. 8

```
° Algorithm GBFM_one_level(w,s,r)
  for (i= 1 to r)
  for (j= 0 to s-1)
  { Consider the a wxw grid superimposed on the bins.
    Shift the grid up and right by s units.
    Perform GBMF within each wxw window
  }
```

FIG. 9

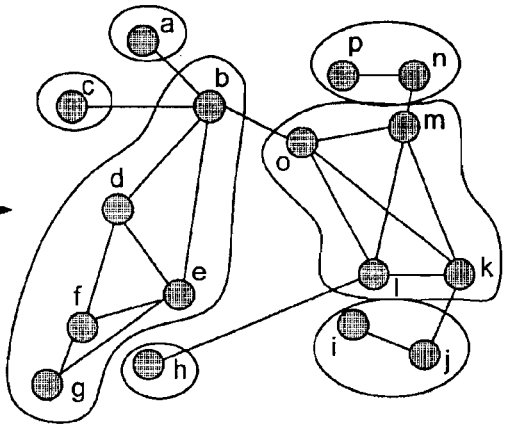
```
° Algorithm dual_GBFM
  for (temp= t down to 1)
  { for (i= 1 to k)  // k should be a small constant like 3
    { while (there exists a region with cost > cost(t) or
      { Reg = a region randomly selected among regions with cost
        Perform quanto-cluster level FM within Reg.
      } // end of while
      while (there exists a region with cost > cost(t) or
      { Reg = a region randomly selected among regions with cost
          Perform gate level FM within Reg.
        } // end of while
      } // end of for i
    } // end of for temp
```

FIG. 10



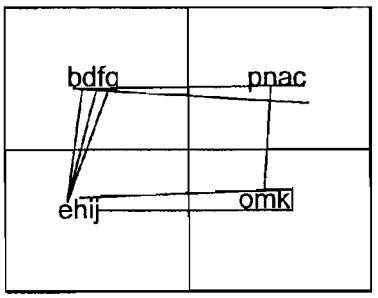
Input circuit

FIG. 11



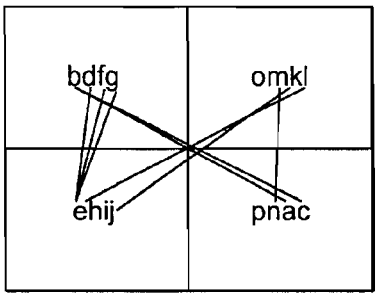
Seven topoclusters

FIG. 12



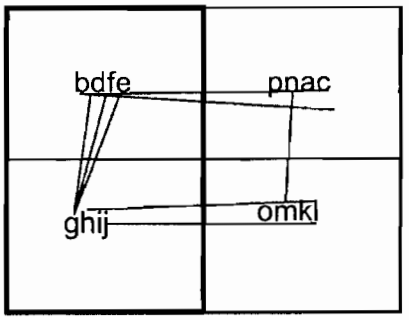
Placement refinement
global move

FIG. 14



Placing topoclusters
in the bins

FIG. 13



Placement refinement
local move

FIG. 15

d	e	a	p
f	b	c	n
g	h	o	m
j	i	l	k

FIG. 16

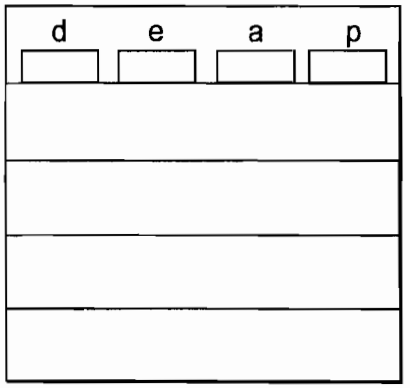


FIG. 17

Done

Router

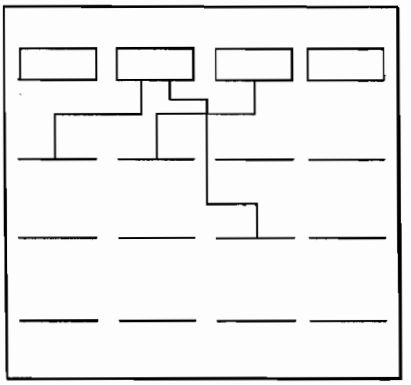


FIG. 18

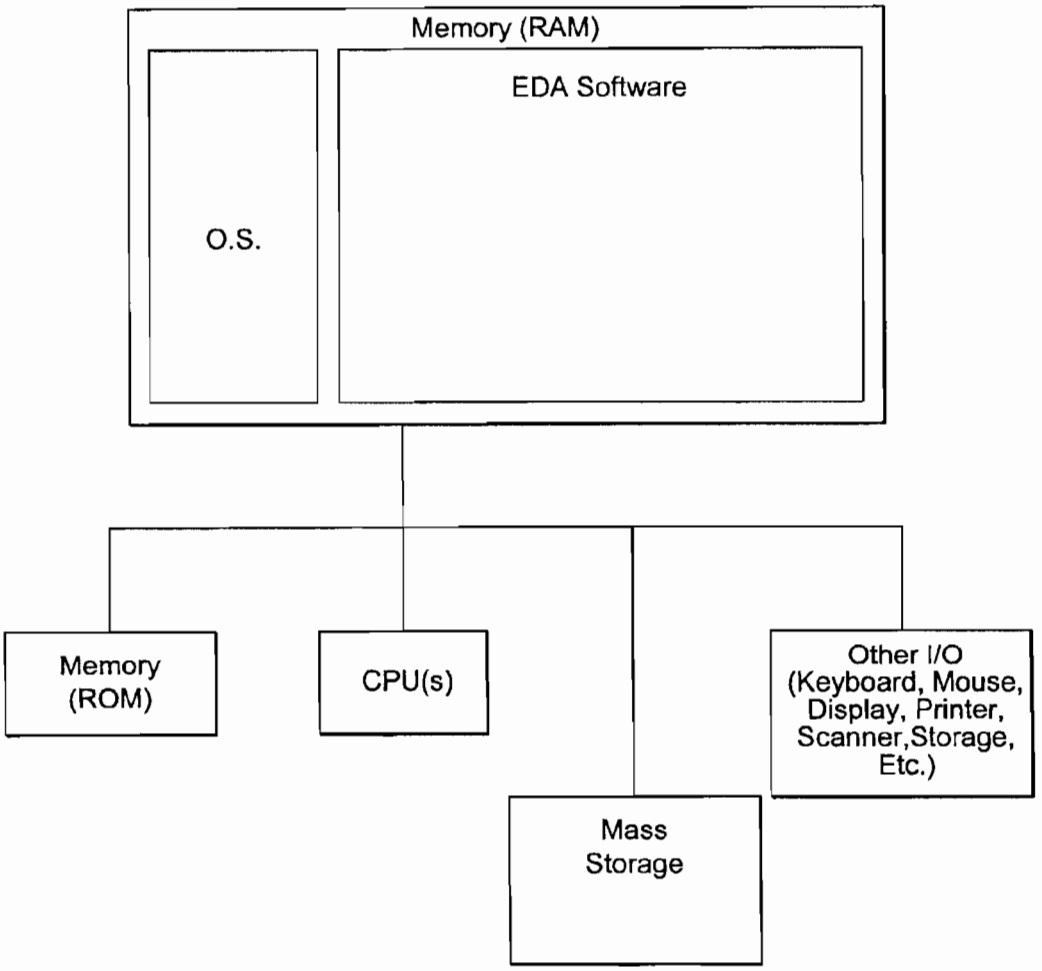

Memory (RAM)
EDA Software
O.S.
Memory (ROM)
CPU(s)
Other I/O (Keyboard, Mouse, Display, Printer, Scanner,Storage, Etc.)
Mass Storage

FIG. 19

PLACEMENT METHOD FOR INTEGRATED CIRCUIT DESIGN USING TOPO-CLUSTERING

This application is related by subject matter to U.S. application Ser. No. 09/097,299 filed on Jun. 12, 1998 entitled METHOD FOR DESIGN OPTIMIZATION USING LOGICAL AND PHYSICAL INFORMATION, filed on even date herewith and incorporated herein by reference an now U.S. Pat. No. 6,286,128.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to placement in integrated circuit design.

2. State of the Art

The physical design of integrated circuits involves placement of gates within a design layout, representing a physical integrated circuit, and routing of interconnections (nets) between gates. The logical integrated circuit design is represented in the form of a netlist, i.e., a list of gates or collections of gates (macros) and nets interconnecting them. A graph representation of a netlist is shown in FIG. 1A. Through placement and routing, the logical integrated circuit design is translated to a physical integrated circuit design. Placement and routing are performed using Electronic Design Automation (EDA) software tools running on powerful computers.

Placement and routing are closely inter-related. As integration density increases, the sheer size of integrated circuit designs challenges current methods of physical design. Furthermore, physical design is required to be more exacting in order to avoid deleterious interactions and to ensure that all design constraints are met.

A number of approaches to the placement problem have been proposed, including simulated annealing, genetic algorithms, mathematical/linear programming, bisection type approaches, etc. One widely-practiced partitioning algorithm known as FM after its originators Fiduccia and Matheyses, is used as the basis for many placement algorithms. In FM, groups of features are formed, and features are exchanged between the groups so as to minimize the number of nets extending between the groups. The FM technique, an example of a module partitioning heuristic, may be represented as follows:

1. Determine initial partition of modules.
2. Loop until no more improvement in the partition results, or until a maximum number of passes is tried:
   a. Free all modules and compute module gains.
   b. Loop while there remains a free module that can be moved:
      i. Select next module to be moved (select free module of maximum gain, subject to area-balance criterion).
      ii. Move selected module to opposite side of partition.
      iii. Update module gains.
   c. Reconstruct best partition of pass.

Gain refers to decrease in the number of nets crossing between opposite sides of the partition.

A major shortcoming of the foregoing technique, as well as other similar techniques, is that after a partition has been made, it is difficult or impossible for gates or modules to cross the partition boundary. This restriction often results in inferior placements. A cycling and overlapping partitioning process is described in Huang and Kahng, Partitioning-Based Standard Cell with an Exact Objective, *Proc. of the International Symposium on Physical Design,* April 1997. This approach, to a limited extent, does allow gates and modules to cross partition boundaries. However, the approach is not cluster-based (is slow) and does not exploit the full power of cycling and overlapping partitioning (produces less-than-adequate quality).

In short, none of these existing placement techniques appears well-equipped to meet the challenges of the deep sub-micron era.

SUMMARY OF THE INVENTION

The present invention, generally speaking, provides a placement method for the physical design of integrated circuits in which natural topological feature clusters (topo-clusters) are discovered and exploited during the placement process. Topo-clusters may be formed based on various criteria including, for example, functional similarity, proximity (in terms of number of nets), and genus. Genus refers to a representation of a netlist in terms of a number of planar netlists—netlists in which no crossing of nets occurs. Topo-clusters drive initial placement, with all of the gates of a topo-cluster being placed initially in a single bin of the placement layout or within a group of positionally-related bins. The portion of a topo-cluster placed within a given bin is called a quanto-cluster. An iterative placement refinement process then follows, using a technique referred to herein as Geometrically-Bounded FM (GBFM), and in particular Dual GBFM. In GBFM, FM is applied on a local basis to windows encompassing some number of bins. From iteration to iteration, windows may shift position and vary in size. When a region bounded by a window meets a specified cost threshold in terms of a specified cost function, that region does not participate. The cost function takes account of actual physical metrics—delay, area, congestion, power, etc. "Dual" refers to the fact that each iteration has two phases. During a first phase, FM is performed within a region on a quanto-cluster basis. During a second phase, FM is performed within the region on a gate basis. GBFM occurs in the context of recursive quadrisection. Hence, after GBFM has been completed, a further quadrisection step is performed in which each bin is divided into four bins, with a quarter of the gates of the original bin being placed in the center of each of the resulting bins. GBFM then follows, and the cycle repeats until each bin contains a fairly small number of gates. Following the foregoing global placement process, the circuit is then ready for detailed placement in which cells are assigned to placement rows.

BRIEF DESCRIPTION OF THE DRAWING

The present invention may be further understood from the following description in conjunction with the appended drawing. In the drawing:

FIG. **1A** is a diagram representing a nedlist;

FIG. **1B** is a diagram in which topo-clusters have been identified within the netlist of FIG. **1A**;

FIG. **1C** is a course netlist in which the topo-clusters of FIG. **1C** are each represented as a single feature;

FIG. **2** is a diagram representing a netlist with genus **3**;

FIG. **3** is a simplified diagram representing an initial placement of topo-clusters on a placement layout, resulting in quanto-clusters;

FIG. **4A** is a diagram showing a first set of windows used to perform GBFM;

FIG. **4B** is a diagram showing a second shifted set of window;

FIG. **5** is a diagram representing a quadrisection process;

FIG. **6** is a diagram representing quadrisection in greater detail;

FIG. **7** is a diagram representing detailed placement;

FIG. **8** is a flow diagram illustrating the placement process;

FIG. **9** is a pseudo-code representation of a placement refinement process;

FIG. **10** is a pseudo-code representation of a placement refinement process in accordance with an alternative embodiment,

FIG. **11** is a graph representing an example netlist;

FIG. **12** is a topo-cluster diagram based on the graph of FIG. **11**;

FIG. **13** is a diagram of quanto-clusters formed from the topo-clusters of FIG. **12**;

FIG. **14** is a diagram showing the results of a global move during placement refinement;

FIG. **15** is a diagram showing the results of a local move during placement refinement;

FIG. **16** is a diagram showing the results of quadrisection following the move of FIG. **15**;

FIG. **17** is a diagram showing row placement of cells represented by the nodes a–k;

FIG. **18** is a diagram of a final routed integrated circuit produced by the foregoing steps;

FIG. **19** is a block diagram of a computer system that may be used to practice the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present placement method is guided by a number of important decisions that contribute to the overall strength of the placement method. First, the placement method is based on clustering. Present day designs are too large to be considered in flat mode. Gates must therefore be clustered to reduce the design space. Second, the placement method is based on quadrisection techniques. Quadrisection techniques are extremely fast as compared to annealing or mathematical programming methods. Quadrisection, as opposed to bisection, better models the two-dimensional nature of the placement problem. Third, the placement technique allows for gates to cross quadrisection boundaries.

Referring to FIG. **8**, a flow diagram of the present placement method is shown. The input to the placement process is assumed to be a netlist. In Step **801**, clustering is performed on the netlist, creating topo-clusters. Then a layout layout is formed as an array of b×b "bins." The number of bins b is determined based on technology-specific considerations (**803**). A 2-D ordering of topo-clusters is then performed, followed by mapping of topo-clusters into "quanto-clusters" (**805**). A quanto-cluster is a bin-size portion of a topo-cluster, and may include the entire topo-cluster. A cycling and overlapping partitioning process is then performed (**807**), referred to herein as Geometrically Bounded FM, or GBFM, described in greater detail hereinafter. Logic optimization may optionally be performed (**809**), after which quadrisection is performed (**811**). During quadrisection, each bin is divided into four bins one-fourth the size. A loop then ensues of GBFM (**813**), optional logic optimization (**815**), and quadrisection (**811**), until the bins each contain a relatively small number of gates, e.g. ten or a few tens of gates. Partitioning is then discontinued (**817**). Nested within this outer loop is an inner loop **819** within which GBFM may be performed repeatedly.

The concept of clustering is illustrated in FIG. **1B**. During clustering, topologically-related circuit elements are grouped together. In FIG. **1B**, three clusters are shown, clusters **101**, **103**, and **105**. Following clustering, the circuit may be represented in the form of a coarse netlist in which clusters are elemental and are joined by inter-cluster nets.

Clustering may be accomplished by various techniques including, for example, techniques based on functional similarity, netlist distance, and genus analysis. Each of these variations will be described in turn.

The nature of netlists makes clustering based on functional similarity quite straightforward. Each cell instance is given a unique name. Related cell instances are given cell names that are quite evidently related. For example, a register might be composed of some number of flip-flops. These flip-flops may have the instance names top/u**1**/registers/control_ff[**7**], top/u**1**/registers/control_ff[**6**], . . . , top/u**1**/registers/control_ff[**0**]. A netlist parser may readily form clusters based on instance name relatedness. Alternatively, information concerning functional relatedness may be determined during logic synthesis based on a high level (e.g., Verilog, VHDL) description and preserved within the netlist format itself. The latter approach is preferable in that functional relatedness is best determined based on a high-level description, but may not be possible if the input netlist does not already include information concerning functional relatedness.

In distance-based clustering, whether two gates belong to the same cluster is determined based on a distance measure. The distance of two gates in a circuit can be defined as the minimum number of nets (or gates) visited in going from the first gate to the second. A bottom-up clustering method is used. Initially, each circuit element is its own cluster. Clusters are then merged to form larger clusters, based on distance considerations.

Clustering may also be performed based on hyper-graph analysis of the netlist graph. A hyper-graph is a graph in which each edge of the graph may have multiple endpoints. A planar graph is one in which no graph edges cross. The genus of a hyper-graph is the number of planar sub-graphs needed to represent it. FIG. **2** shows a representation of a netlist having genus **3**. Genus analysis of a hyper-graph may be performed as follows. First a maximal planar netlist of the original netlist is obtained. This planar netlist is said to be on plane **1**. The planar netlist is deleted and a new planar netlist is obtained. The second planar netlist is said to be on plane **2**. This process repeats until all nets have been removed. The total number of planes obtained is called the genus of the (original) netlist. This process provides insight into the complexity of the netlist. Furthermore, regions that are highly non-planar (as determined by the above analysis) can be clustered together.

Clusters formed by any of the foregoing methods, or by other methods that take into account circuit topology, are referred to as natural clusters or topological (topo) clusters.

The identification of clusters may involve trial and error. The following cost function may be used in evaluating the quality of a topo-clustering:

$$Kij=(Pij-1)/(Ti-1)$$

where Kij is the "credit" of net Ni in cluster j, Pij is the total number of terminals of net Ni in clusterj and Ti is the total

number of terminals of net i. The absorption cut benefit, K, is defined as the summation of Kij over all nets and all topo-clusters. If m denotes the number of topo-clusters, a good topo-clustering is one with large values of K and m.

The usefulness of topo-clusters is evident in both obtaining a good initial placement of circuit elements and in performing placement refinement. Using topo-clusters, the initial placement may be expected to have a significantly lower cost function (and hence be significantly closer to the final placement) than if topo-clusters are not used. Also, using topo-clusters, the cost function of the placement may be reduced in a more computationally-efficient manner.

In one exemplary embodiment, initial placement is performed by, beginning in the center of the design layout, using the bins in a predetermined spiral order to place each cluster in turn in as many bins as required by the cluster, as shown in FIG. **3**. In placing topo-clusters, the topo-clusters become "bin-quantized" to form quanto-clusters. In an alternative embodiment, initial placement is performed in serpentine fashion, e.g., in row-major order for a first row of bins, reverse-row-major order for a succeeding row of bins, then row-major order again, etc. Preferably, topo-clusters are not placed in random order but rather are ordered based on a measure of the inter-relatedness of different topo-clusters.

Following initial placement, placement refinement occurs. Placement refinement is performed iteratively, each iteration involving quadrisection followed by a variant of FM, referred to herein as Dual GBFM. "Dual" refers to the fact that moves are performed first at the quanto-cluster level and then at the gate level. Dual GBFM differs from conventional FM in numerous respects, including the following:

1. Initial placement has already been performed. Moves may therefore be evaluated based on an actual physical cost function, e.g., one that measures congestion (routability), area, timing improvement, power, etc.
2. GBFM is applied selectively (by region) and iteratively. Conventional FM, in general, is applied universally and recursively.
3. Moves include cluster-level moves, followed by gate-level moves.
4. GBFM is multi-way. Although the possibility of multi-way FM has been recognized, virtually all commercial applications of FM have been two-way because of the large computational cost of multi-way FM. Using a shifting-window approach, GBFM achieves a comparable effect as large multi-way FM but in a way that is computationally manageable.

The GBFM process allows cells assigned to one partition to freely move to another partition using a controlled mechanism. GBFM, in accordance with exemplary embodiments, is described by the pseudo-code routines of FIG. **9** and FIG. **10**. Referring first to FIG. **9**, at a particular level of quadrisection, a window-pane overlay is applied to the design layout, as illustrated in FIG. **4**A. GBFM is performed within each window. GBFM is multi-way FM using a geometric cost function, i.e., a cost function that is a function of distance, area, etc., not merely a function of "cuts" (topology). The windows are then redefined by shifting the window-pane overlay such that the new windows partially overlap the previous windows, as illustrated in FIG. **4**B, and the process is repeated. The parameters w, s and r define a window size, a shift amount, and the number of repetitions (or rounds), respectively. In early stages of the placement process, the value of w should be large, and r should also be large. In later stages of the placement process, w should be small, s can be larger, and r should be small. The three parameters may be adjusted as a function of the given netlist, based on topo-clusters. For example, if analysis reveals a large number of disconnected topo-clusters, then w may be small, s may be large and r may be small. The large number of disconnected topo-clusters makes the problem easier, hence the window size may be reduced, saving work for the algorithm. In general, the parameters w, s and r may be determined empirically for different types of circuit layout problems; thereafter, the algorithm may be "tuned" for different circuit layout problems by adjusting the parameters w, s and r.

GBFM operates both on a quanto-cluster (bin) basis (first half of GBFM) and on a gate basis (second half of GBFM).

Referring to FIG. **10**, in accordance with an alternative embodiment, GBFM uses a notion of temperature (as in simulated annealing, for example). The result of introducing the notion of temperature is that initially only moves that will result in fairly substantial improvement in the cost function are considered (high temperature, high cost threshold). In later stages, moves that will result in more modest improvement are considered (low temperature, low cost threshold). Instead of considering a large number of regions (windows) systematically and exhaustively, regions are randomly selected based on temperature and cost. Computational savings may therefore result. In an exemplary embodiment, at each temperature, a small number of iterations of GBFM are performed.

Following GBFM, quadrisection is again performed as illustrated in FIG. **5**. During quadrisection, the circuit elements are divided into fourths, with one fourth of the circuit elements being placed in each of four new bins, as illustrated in FIG. **6**. The circuit elements are placed in the center of the new bins. Quadrisection is followed again by GBFM This process repeats until each bin contains a small number of gates, e.g. ten or a few tens of gates.

Following the foregoing global placement process, the circuit is then ready for detailed placement in which cells are assigned to placement rows as illustrated in FIG. **7**.

The foregoing process may be more fully understood with reference to a specific example. Referring to FIG. **11**, an example netlist is shown as represented by a graph. Each of the nodes a-p represents a cell. As a preliminary step, clustering is performed based on any of the described techniques or other suitable clustering techniques. In the present example, clustering is assumed to result in seven topo-clusters as shown in FIG. **12**. Topo-clusters are then placed in bins defined as part of the design layout. For example, referring to FIG. **13**, the topo-cluster omkl is placed in the upper right-hand bin, forming its own quanto-cluster. Part of the topo-cluster bdefg is placed in the upper left-hand bin, with the remainder of the topo-cluster (e) being placed in the lower left-hand bin. The cells bdefg therefore form a quanto-cluster. The topo-clusters h and ij are also placed in the lower left-hand bin. The cells ehij form another quanto-cluster. Finally, the topo-clusters pn, a and c are placed in the lower right-hand bin, forming a quanto-cluster. Initially, all of the cells in a bin are placed in the center of the bin.

Placement refinement then ensues, beginning with global moves, i.e., exchange of quanto-clusters, followed by local moves, i.e., exchange of gates. Global moves and local moves are applied within regions (windows) as previously described. In FIG. **14**, the window encompasses the upper and lower right-hand bins. Within this window, exchange of the quanto-clusters omkl and pnac results in a cost improvement. In FIG. **15**, at a later stage, during a local move phase, the window encompasses the upper and lower left-hand

bins. Within this window, exchange of the cells e and g results in a cost improvement.

Following GBFM, quadrisection then follows. Each bin is divided into four smaller bins and the cells within the bin are divided into four groups, each group being parcelled out to a different one of the new smaller bins. In the simplified example, as shown in FIG. **16**, quadrisection results in a single cell being placed in the center of each of the resulting cells. The iteration of GBFM and quadrisection therefore concludes, although in the chosen example only a single iteration occurred. In an actual example, a large number of iterations may be expected to occur. The cells are then placed within placement rows (FIG. **17**) and subsequently routed (FIG. **18**).

The present invention may be embodied in various forms, including computer-implemented methods, computer systems configured to implement such methods, computer-readable media containing instructions for implementing such methods, etc. Examples of computer-implemented methods embodying the invention have been described. Reducing such methods to tangible form as computer-readable media may be accomplished by methods well-known in the art.

Referring to FIG. **19**, a diagram is shown of a computer system that may be used to practice the present invention. Attached to a system bus are one or more CPUs, read-only memory (ROM), read/write memory (RAM), mass storage, and other I/O devices. The other I/O devices will typically include a keyboard, a pointing device, and a display, and may further include any of a wide variety of commerically-available I/O devices, including, for example, magnetic storage devices, optical storage devices, other storage devices, printers, etc. Stored within memory (e.g., RAM) is EDA software implementing methods of the type previously described. The EDA software.

It will be appreciated by those of ordinary skill in the art that the invention can be embodied in other specific forms without departing from the spirit or essential character thereof. The presently disclosed embodiments are therefore considered in all respects to be illustrative and not restrictive. The scope of the invention is indicated by the appended claims rather than the foregoing description, and all changes which come within the meaning and range of equivalents thereof are intended to be embraced therein.

It will be appreciated by those of ordinary skill in the art that the invention can be embodied in other specific forms without departing from the spirit or essential character thereof The presently disclosed embodiments are therefore considered in all respects to be illustrative and not restrictive. The scope of the invention is indicated by the appended claims rather than the foregoing description, and all changes which come within the meaning and range of equivalents thereof are intended to be embraced therein.

What is claimed is:

1. A method of placing circuit elements on an integrated circuit design layout comprising the steps of:

grouping circuit elements into clusters based on topological relatedness of the circuit elements of a cluster;

placing circuit elements by cluster within bins defined on the circuit design layout wherein, for a first cluster, said placing circuit elements by cluster includes repeatedly placing circuit elements from said first cluster in to a centralized bin until the centralized bin reaches capacity and placing remaining circuit elements from said first cluster into a bin which is located in a predetermined spiral order to the centralized bin until all circuit elements of said first cluster have been placed;

defining a continuous region specified by said bins; and

applying a placement refinement technique to said continuous region to produce a placement that is improved as measured by a cost function.

2. The method of claim **1**, wherein said placement refinement technique comprises moving at least one of circuit elements and clusters between bins within a region.

3. The method of claim **2**, wherein within at least some regions both circuit elements and clusters are moved between bins.

4. The method of claim **3**, wherein clusters are moved between bins first, and circuit elements are moved between bins afterward.

5. The method of claim **3**, comprising the further steps of:

shifting a set of windows by a program-controlled shift amount relative to the integrated circuit design layout to thereby define a set of different regions; and

applying said placement refinement technique to at least some of said different regions.

6. The method of claim **5**, comprising the further steps of repeating said shifting and applying steps a program-controlled number of times.

7. The method of claim **6**, comprising the further steps of, for each of a plurality of iterations:

defining a plurality of regions using a set of windows of a program-controlled size;

applying said placement refinement technique to at least some of said regions;

shifting the set of windows by a program-controlled shift amount relative to the integrated circuit design layout to thereby define a set of different regions;

applying said placement refinement technique to at least some of said different regions; and

repeating said shifting and applying steps a program-controlled number of times.

8. The method of claim **1**, comprising the further steps of:

quadrisecting said bins;

defining a plurality of different regions; and

again applying said placement refinement technique to at least some of the plurality of different regions.

9. The method of claim **8**, comprising repeating the steps of quadrisection and again applying said placement refinement technique until each bin contains at most a predetermined number of gates.

10. The method of claim **9**, wherein said predetermined number of gates is less than thirty.

11. The method of claim **1**, wherein grouping circuit elements comprises grouping circuit elements by similar function.

12. The method of claim **1**, wherein grouping circuit elements comprises grouping circuit elements by netlist proximity.

13. The method of claim **1**, wherein grouping including the steps of:

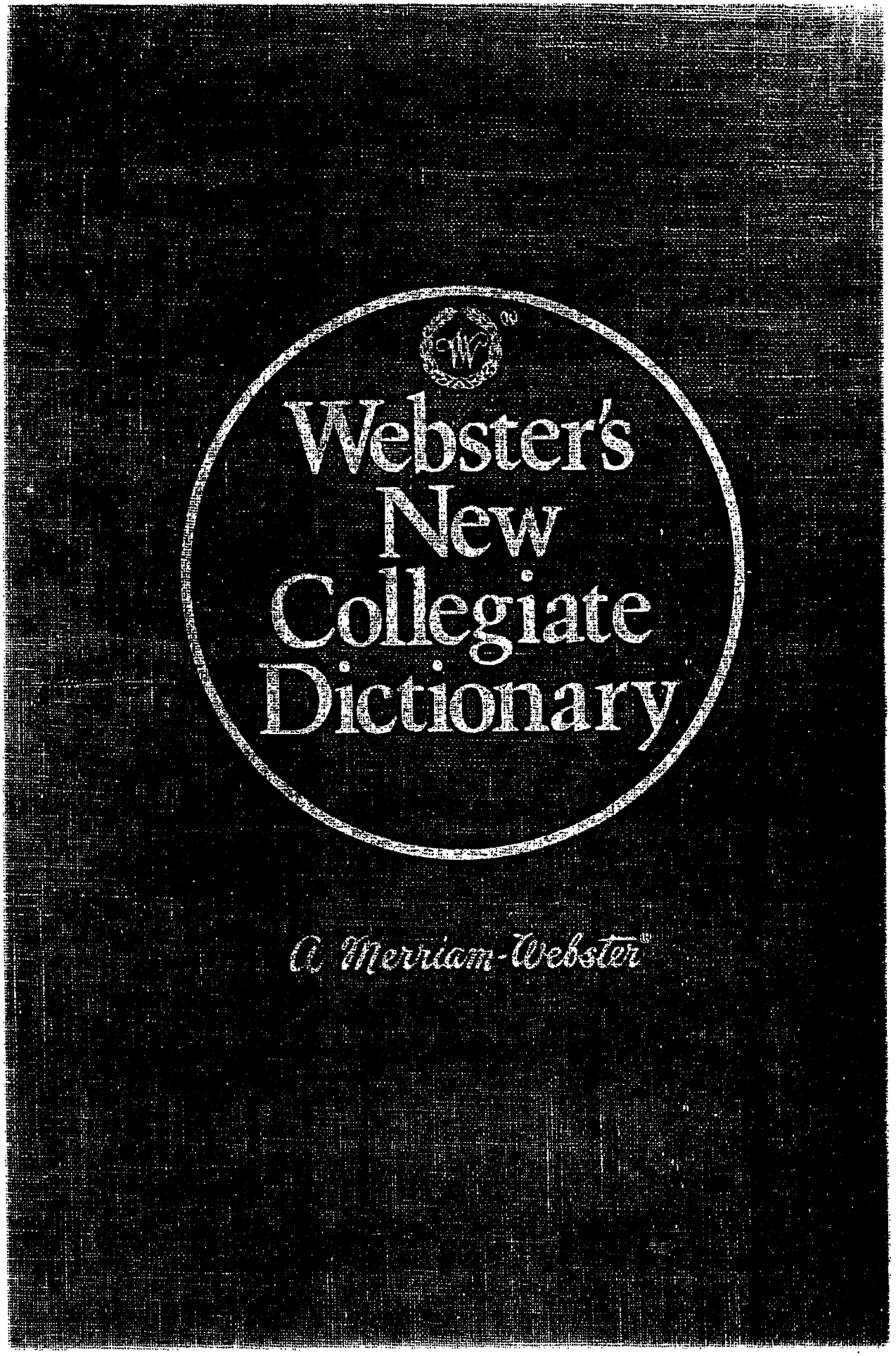
creating a first planar subgraph;

creating a second planar subgraph;

repeating the preceding steps until all the nets have been removed.

* * * * *

TAB 28


Webster's
New
Collegiate
Dictionary
A Merriam-Webster®

New Collegiate Dictionary

A Merriam-Webster®

G. & C. MERRIAM COMPANY
Springfield, Massachusetts, U.S.A.

Library of Congress Cataloging in Publication Data
Main entry under title:

Webster's new collegiate dictionary.

Editions for 1898–1948 have title: Webster's collegiate dictionary.
Includes index.
1. English language—Dictionaries.
PE1628.W4M4 1981 423 80-25144
ISBN 0-87779-408-1
ISBN 0-87779-409-x (indexed)
ISBN 0-87779-410-3 (deluxe)

Made in the United States of America

504948 RMcN8281

Contents

Preface

Webster's New Collegiate Dictionary is a completely new volume in the Merriam-Webster series of dictionaries. It is a general dictionary edited for use in school or college, in the office, and in the home—in short, wherever information about English words is likely to be sought. The average user should rarely have occasion to look for information about the vocabulary of present-day English that is not available within these pages.

The first Merriam-Webster Collegiate appeared in 1898 and quickly won the esteem of student and general reader. A second edition was published in 1910, and subsequent editions came out in 1916, 1931, 1936, 1949, and 1963. This eighth in the series incorporates the best of the time-tested features of its predecessors and introduces new features designed to add to its usefulness. Its more than 1500 pages make it the most comprehensive Merriam-Webster Collegiate ever published.

The heart of Webster's New Collegiate Dictionary is the more than 1300 pages given over to the A–Z vocabulary. The information there set down derives not only from the 10,000,000 citations which were available to the editors of Webster's Third New International Dictionary and the 1963 Collegiate but also from the considerably more than 1,000,000 citations collected since the publication of these books. Thus each entry is based on a constantly updated file of actual English usage.

Those entries known to be trademarks or service marks are so labeled and are treated in accordance with a formula approved by the United States Trademark Association. No entry in this dictionary, however, should be regarded as affecting the validity of any trademark or service mark.

A noteworthy feature of the vocabulary section is the nearly 900 pictorial illustrations, many of which were drawn especially for this book. These illustrations were selected not simply for their decorative function but particularly for their value in clarifying definitions.

The front matter—those pages preceding the A–Z vocabulary—contains two important sections. The Explanatory Notes should be read by every user of the dictionary since a thorough understanding of the information contained in them will contribute markedly to the value of this book. And all users of the dictionary are urged to read the lucid essay on the English language which was written for this Collegiate by Professor W. Nelson Francis of Brown University.

The back matter—those pages following the A–Z vocabulary—contains several sections that dictionary users have long found helpful. These include more than five hundred Foreign Words and Phrases that occur frequently in English texts but that have not become part of the English vocabulary; several thousand proper names that are entered under the separate headings Biographical Names and Geographical Names; and a list of the Colleges and Universities of the United States and Canada. There is also a Handbook of Style in which various stylistic conventions (as of punctuation and capitalization) are concisely summarized.

Webster's New Collegiate Dictionary has been edited by the trained staff of the G. & C. Merriam Co. It is the result of a collaborative effort, and it would be invidious to single out particular editors for special mention. At the same time, it would be ungracious to observe the anonymity which is often the lot of the present-day lexicographer, and so a list of those who contributed substantially to the completion of this book is printed below.

Webster's New Collegiate Dictionary is the product of a company that has been publishing dictionaries for more than 125 years. It is offered to the user with the conviction that it will serve him well.

Henry Bosley Woolf
Editor in Chief

Editorial Staff

heir to receive property as an inheritance **2 :** DEATH TAX; *esp* **:** ESTATE TAX

in-hib-it \in-'hib-ət\ *vb* [ME *inhibiten*, fr. L *inhibitus*, pp. of *inhibēre*, fr. *in-* + *habēre* to have — more at HABIT] *vt* **1 :** to prohibit from doing something **2 a :** to hold in check **:** RESTRAIN **b :** to discourage from free or spontaneous activity esp. through the operation of inner psychological impediments or of social controls ~ *vi* **:** to cause inhibition *syn* see FORBID *ant* allow — **in-hib-i-tive** \-ət-iv\ *adj* — **in-hib-i-to-ry** \-ə-ˌtōr-ē, -ˌtȯr-\ *adj*

in-hi-bi-tion \ˌin-(h)ə-'bish-ən\ *n* **1 a :** the act of inhibiting **:** the state of being inhibited **b :** something that forbids, debars, or restricts **2 :** an inner impediment to free activity, expression, or functioning: as **a :** a psychical activity imposing restraint upon another activity **b :** a restraining of the function of a bodily organ or an agent (as an enzyme)

in-hib-i-tor *or* **in-hib-it-er** \in-'hib-ət-ər\ *n* **:** one that inhibits; *esp* **:** an agent that slows or interferes with a chemical action (as rusting)

in-hos-pi-ta-ble \ˌin-(ˌ)häs-'pit-ə-bəl, (ˌ)in-'häs-(ˌ)pit-\ *adj* **1 :** not showing hospitality **:** not friendly or receptive **2 :** providing no shelter or sustenance **:** BARREN — **in-hos-pi-ta-ble-ness** *n* — **in-hos-pi-ta-bly** \-blē\ *adv*

in-hos-pi-tal-i-ty \(ˌ)in-ˌhäs-pə-'tal-ət-ē\ *n* **:** the quality or state of being inhospitable

in-house \ˌin-ˌhaüs, 'in-'\ *adj* **:** of, relating to, or carried on within a group or organization ⟨can be an outside or an ~ job —*Book Production Industry*⟩ — **in-house** *adv*

in-hu-man \(ˌ)in-'hyü-mən, -'yü-\ *adj* [MF & L; MF *inhumain*, fr. L *inhumanus*, fr. *in-* + *humanus* human] **1 a :** lacking pity, kindness, or mercy **:** SAVAGE ⟨an ~ tyrant⟩ **b :** COLD, IMPERSONAL ⟨his usual quiet, almost ~ courtesy —F. Tennyson Jesse⟩ **c :** not worthy of or conforming to the needs of human beings ⟨~ living conditions⟩ **2 :** of or suggesting a nonhuman class of beings — **in-hu-man-ly** *adv* — **in-hu-man-ness** \-mən-nəs\ *n*

in-hu-mane \ˌin-(ˌ)hyü-'mān, -(ˌ)yü-\ *adj* [MF *inhumain* & L *inhumanus*] **:** not humane **:** INHUMAN 1 — **in-hu-mane-ly** *adv*

in-hu-man-i-ty \-'man-ət-ē\ *n, pl* **-ties 1 a :** the quality or state of being cruel or barbarous **b :** a cruel or barbarous act **2 :** absence of warmth or geniality **:** IMPERSONALITY

in-hume \in-'hyüm\ *vt* **in-humed; in-hum-ing** [prob. fr. F *inhumer*, fr. L *inhumare*, fr. *in-* + *humus* earth — more at HUMBLE] **:** BURY, INTER — **in-hu-ma-tion** \ˌin-hyü-'mā-shən\ *n*

in-im-i-cal \in-'im-i-kəl\ *adj* [LL *inimicalis*, fr. L *inimicus* enemy — more at ENEMY] **1 a :** having the disposition of an enemy **:** HOSTILE **b :** reflecting or indicating hostility **:** UNFRIENDLY **2 :** being adverse usu. by reason of hostility or malevolence *syn* see ADVERSE — **in-im-i-cal-ly** \-i-k(ə-)lē\ *adv*

in-im-i-ta-ble \(ˌ)in-'im-ət-ə-bəl\ *adj* [MF or L; MF, fr. L *inimitabilis*, fr. *in-* + *imitabilis* imitable] **:** not capable of being imitated **:** MATCHLESS — **in-im-i-ta-ble-ness** *n* — **in-im-i-ta-bly** \-blē\ *adv*

in-iq-ui-tous \in-'ik-wət-əs\ *adj* **:** characterized by iniquity *syn* see VICIOUS *ant* righteous — **in-iq-ui-tous-ly** *adv* — **in-iq-ui-tous-ness** *n*

in-iq-ui-ty \-wət-ē\ *n, pl* **-ties** [ME *iniquite*, fr. MF *iniquité*, fr. L *iniquitat-*, *iniquitas*, fr. *iniquus* uneven, fr. *in-* + *aequus* equal] **1 :** gross injustice **:** WICKEDNESS **2 :** an iniquitous act or thing **:** SIN

[1]ini-tial \in-'ish-əl\ *adj* [MF & L; MF, fr. L *initialis*, fr. *initium* beginning, fr. *initus*, pp. of *inire* to go into, fr. *in-* + *ire* to go — more at ISSUE] **1 :** of or relating to the beginning **:** INCIPIENT **2 :** placed at the beginning **:** FIRST — **ini-tial-ly** \-'ish-(ə-)lē\ *adv* — **ini-tial-ness** \-'ish-əl-nəs\ *n*

[2]initial *n* **1 a :** the first letter of a name **b** *pl* **:** the first letter of each word in a full name ⟨found that their ~*s* were identical⟩ **2 :** a large letter beginning a text or a division or paragraph **3 :** ANLAGE, PRECURSOR: *specif* **:** a meristematic cell

[3]initial *vt* **ini-tialed** *or* **ini-tialled; ini-tial-ing** *or* **ini-tial-ling** \-'ish-(ə-)liŋ\ **1 :** to affix an initial to **2 :** to authenticate or give preliminary approval to by affixing the initials of an authorizing representative

ini-tial-ism \in-'ish-ə-ˌliz-əm\ *n* **:** an acronym formed from initial letters

ini-tial-ize \-ˌlīz\ *vt* **-ized; -iz-ing :** to set (as a computer program counter) to a starting position or value — **ini-tial-iza-tion** \-in-ˌish-(ə-)lə-'zā-shən\ *n*

initial rhyme *n* **1 :** ALLITERATION **2 :** BEGINNING RHYME

initial side *n* **:** the stationary straight line that contains the point about which another straight line is revolved in forming a trigonometric figure

initial teaching alphabet *n* **:** a 44-symbol alphabet designed esp. for use in the initial stages of teaching children to read English

[1]ini-ti-ate \in-'ish-ē-ˌāt\ *vt* **-at-ed; -at-ing** [LL *initiatus*, pp. of *initiare*, fr. L, to induct, fr. *initium*] **1 :** to cause or facilitate the beginning of **:** set going ⟨~ a program of reform⟩ ⟨enzymes that ~ fermentation⟩ **2 :** to instruct in the rudiments or principles of something **:** INTRODUCE **3 :** to induct into membership by or as if by special rites *syn* see BEGIN *ant* consummate — **ini-ti-a-tor** \-ˌāt-ər\ *n*

[2]ini-tiate \in-'ish-(ē-)ət\ *adj* **1 a :** initiated or properly admitted (as to membership or an office) **b :** instructed in some secret knowledge **2** *obs* **:** relating to an initiate

[3]ini-tiate \in-'ish-(ē-)ət\ *n* **1 :** a person who is undergoing or has undergone an initiation **2 :** a person who is instructed or adept in some special field

ini-ti-a-tion \in-ˌish-ē-'ā-shən\ *n* **1 a :** the act or an instance of initiating **b :** the process of being initiated **c :** the rites, ceremonies, ordeals, or instructions with which one is made a member of a sect or society or is invested with a particular function or status **2 :** the condition of being initiated into some experience or sphere of activity **:** KNOWLEDGEABLENESS ⟨clear to a reader of any degree of ~ —J. W. Beach⟩

[1]ini-tia-tive \in-'ish-ət-iv\ *adj* **:** of or relating to initiation **:** INTRODUCTORY, PRELIMINARY

[2]initiative *n* **1 :** an introductory step ⟨he took the ~ in attempting to settle the issue⟩ **2 :** energy or aptitude displayed in initiation of action **:** ENTERPRISE ⟨a man of great ~⟩ **3 a :** the right to initiate legislative action **b :** a procedure enabling a specified number of voters by petition to propose a law and secure its submission to the electorate or to the legislature for approval — compare REFERENDUM — **on one's own initiative :** at one's own discretion **:** independently of outside influence or control

ini-tia-to-ry \in-'ish-(ē-)ə-ˌtōr-ē, -ˌtȯr-\ *adj* **1 :** constituting a beginning **2 :** tending or serving to initiate

in-ject \in-'jekt\ *vt* [L *injectus*, pp. of *inicere*, fr. *in-* + *jacere* to throw — more at JET] **1 a :** to throw, drive, or force into something ⟨~ fuel into an engine⟩ **b :** to force a fluid into (as for medical purposes) **2 :** to introduce as an element or factor in or into some situation or subject ⟨condemning any attempt to ~ religious bigotry into the campaign —*Current Biog.*⟩ — **in-ject-able** \-'jek-tə-bəl\ *adj* — **in-jec-tor** \-'jek-tər\ *n*

in-jec-tant \-'jek-tənt\ *n* **:** a substance that is injected into something

in-jec-tion \in-'jek-shən\ *n* **1 a :** an act or instance of injecting ⟨the ~ of academic values into that aspect of our national life is a highly desirable . . . development —Goodwin Watson⟩ **b :** the placing of an artificial satellite or a spacecraft into an orbit or on a trajectory; *also* **:** the time or place at which injection occurs **2 :** something (as a medication) that is injected

in-jec-tor razor \in-ˌjek-tər-\ *n* **:** a safety razor with a narrow single-edged blade that is forced into place by a blade dispenser

in-ju-di-cious \ˌin-jü-'dish-əs\ *adj* **:** not judicious **:** INDISCREET, UNWISE — **in-ju-di-cious-ly** *adv* — **in-ju-di-cious-ness** *n*

in-junc-tion \in-'jəŋ(k)-shən\ *n* [MF & LL; MF *injonction*, fr. LL *injunction-*, *injunctio*, fr. L *injunctus*, pp. of *injungere* to enjoin — more at ENJOIN] **1 :** the act or an instance of enjoining **:** ORDER, ADMONITION **2 :** a writ granted by a court of equity whereby one is required to do or to refrain from doing a specified act — **in-junc-tive** \-'jəŋ(k)-tiv\ *adj*

in-jure \'in-jər\ *vt* **in-jured; in-jur-ing** \'inj-(ə-)riŋ\ [back-formation fr. *injury*] **1 a :** to do an injustice to **:** WRONG **b :** to harm, impair, or tarnish the standing of **c :** to give pain to ⟨~ a man's pride⟩ **2 a :** to inflict bodily hurt on **b :** to impair the soundness of **c :** to inflict material damage or loss on — **in-jur-er** \'in-jər-ər\ *n*

syn INJURE, HARM, HURT, DAMAGE, IMPAIR, MAR *shared meaning element* **:** to affect injuriously *ant* aid

in-ju-ri-ous \in-'jůr-ē-əs\ *adj* **1 :** inflicting or tending to inflict injury **:** DETRIMENTAL ⟨~ to health⟩ **2 :** ABUSIVE, DEFAMATORY ⟨speak not ~ words —George Washington⟩ — **in-ju-ri-ous-ly** *adv* — **in-ju-ri-ous-ness** *n*

in-ju-ry \'inj-(ə-)rē\ *n, pl* **-ries** [ME *injurie*, fr. L *injuria*, fr. *injurus* injurious, fr. *in-* + *jur-*, *jus* right — more at JUST] **1 a :** an act that damages or hurts **:** WRONG **b :** violation of another's rights for which the law allows an action to recover damages **2 :** hurt, damage, or loss sustained *syn* see INJUSTICE

in-jus-tice \(ˌ)in-'jəs-təs\ *n* [ME, fr. MF, fr. L *injustitia*, fr. *injustus* unjust, fr. *in-* + *justus* just] **1 :** absence of justice **:** violation of right or of the rights of another **:** UNFAIRNESS **2 :** an unjust act *syn* INJUSTICE, INJURY, WRONG, GRIEVANCE *shared meaning element* **:** an act that inflicts undeserved hurt

[1]ink \'iŋk\ *n, often attrib* [ME *enke*, fr. OF, fr. LL *encaustum*, fr. neut. of L *encaustus* burned in, fr. Gk *enkaustos*, verbal of *enkaiein* to burn in — more at ENCAUSTIC] **1 :** a colored usu. liquid material for writing and printing **2 :** the black protective secretion of a cephalopod — **ink-i-ness** \'iŋ-kē-nəs\ *n* — **inky** \'iŋ-kē\ *adj*

[2]ink *vt* **:** to put ink on ⟨~ a pen⟩; *also* **:** to write on, draw, or sign in ink ⟨~*ed* a new contract⟩

ink-ber-ry \'iŋk-ˌber-ē\ *n* [fr. the use of the berries for making ink] **1 a :** a holly (*Ilex glabra*) of eastern No. America with evergreen oblong leathery leaves and small black berries **b :** POKEWEED **2 :** the fruit of an inkberry

ink-blot test \ˌiŋk-ˌblät-\ *n* **:** any of several psychological tests based on the interpretation of irregular figures (as blots of ink)

[1]ink-horn \'iŋk-ˌhȯ(ə)rn\ *n* **:** a small portable bottle (as of horn) for holding ink

[2]inkhorn *adj* **:** ostentatiously learned **:** PEDANTIC ⟨~ terms⟩

in-kle \'iŋ-kəl\ *n* [origin unknown] **:** a colored linen tape or braid woven on a very narrow loom and used for trimming; *also* **:** the thread used

in-kling \'iŋ-kliŋ\ *n* [ME *yngkilling*, prob. fr. *inclen* to hint at; akin to OE *incan* suspicion, Lith *ingis* sluggard] **1 :** a slight indication or suggestion **:** HINT, CLUE ⟨there was no path — no ~ even of a track —*New Yorker*⟩ **2 :** a slight knowledge or vague notion ⟨had not the faintest ~ of what it was all about —H. W. Carter⟩

ink-stand \'iŋk-ˌstand\ *n* **:** INKWELL; *also* **:** a stand with fittings for holding ink and pens

ink-well \'iŋ-ˌkwel\ *n* **:** a container (as in a school desk) for ink

inky cap *n* **:** a mushroom (genus *Coprinus*, esp. *C. atramentarius*) whose pileus melts into an inky fluid after the spores have matured

inky cap

in-laid \'in-ˌlād\ *adj* **1 a :** set into a surface in a decorative design ⟨tables with ~ marble⟩ **b :** decorated with a design or material set into a surface ⟨a table with an ~ top⟩ **2** *of linoleum* **:** having a design that goes all the way through to the backing

[1]limber \'lim-bər\ *n* [ME *lymour*] : a two-wheeled vehicle to which a gun or caisson may be attached

[2]limber *adj* [origin unknown] 1 : capable of being shaped : FLEXIBLE 2 : having a supple and resilient quality (as of mind or body) : AGILE, NIMBLE — **limber-ly** *adv* — **limber-ness** *n*

[3]limber *vb* **limbered; limber-ing** \-b(ə-)riŋ\ *vt* : to cause to become limber ⟨~ up his fingers⟩ ~ *vi* : to become limber ⟨~ up by running⟩

limbers \'lim-bərz\ *n pl* [modif. of F *lumière*, fr. OF, light, opening, fr. L *luminare* window — more at LUMINARY] : gutters or conduits on each side of the keelson of a ship that provide a passage for water to the pump well

limbic \'lim-bik\ *adj* [NL *limbicus* of a border or margin, fr. L *limbus*] : of, relating to, or being the limbic system of the brain

limbic system *n* : a group of subcortical structures (as the hypothalamus, the hippocampus, and the amygdala) of the brain that are concerned esp. with emotion and motivation

[1]limbo \'lim-(,)bō\ *n, pl* **limbos** [ME, fr. ML, abl. of *limbus* limbo, fr. L, border — more at LIMB] 1 *often cap* : an abode of souls that are according to Roman Catholic theology barred from heaven because of not having received Christian baptism 2 a : a place or state of restraint or confinement b : a place or state of neglect or oblivion ⟨proposals kept in ~⟩ c : an intermediate or transitional place or state

[2]limbo *n, pl* **limbos** [native name in West Indies] : a West Indian acrobatic dance orig. for men that involves bending over backwards and passing under a horizontal pole lowered slightly for each successive pass

Limburger \'lim-,bər-gər\ *n* [Flem, one from Limburg, fr. *Limburg*, Belgium] : a creamy semisoft surface-ripened cheese with a rind of pungent odor

limbus \'lim-bəs\ *n* [L, border] : a border distinguished by color or structure; *esp* : the marginal region of the cornea of the eye by which it is continuous with the sclera

[1]lime \'līm\ *n* [ME, fr. OE *līm*; akin to OHG *līm* birdlime, L *linere* to smear, *levis* smooth, Gk *leios*] 1 : BIRDLIME 2 a : a caustic highly infusible solid that consists of calcium oxide often together with magnesia, that is obtained by calcining forms of calcium carbonate (as shells or limestone), and that is used in building (as in mortar and plaster) and in agriculture — called also *caustic lime* b : a dry white powder consisting essentially of calcium hydroxide that is made by treating caustic lime with water c : CALCIUM ⟨carbonate of ~⟩

[2]lime *vt* **limed; lim-ing** 1 : to smear with a sticky substance (as birdlime) 2 : to entangle with or as if with birdlime 3 : to treat or cover with lime ⟨~ the lawn in the spring⟩

[3]lime *adj* : of, relating to, or containing lime or limestone

[4]lime *n* [alter. of ME *lind*, fr. OE; akin to OHG *linta* linden] : LINDEN 1a

[5]lime *n* [F, fr. Prov *limo*, fr. Ar *līm*] 1 : a spiny tropical citrus tree (*Citrus aurantifolia*) with elliptic oblong narrowly winged leaves 2 : the small globose greenish yellow fruit of a lime with an acid juicy pulp used as a flavoring agent and as a source of vitamin C

limeade \līˈmād\ *n* : a beverage of sweetened lime juice mixed with plain or carbonated water

lime glass *n* : glass containing a substantial proportion of lime

lime-juicer \'līm-,jü-sər\ *n* [fr. the use of lime juice on British ships as a beverage to prevent scurvy] 1 *slang* a : a British ship b : a British sailor 2 *slang* : ENGLISHMAN

lime-kiln \-,kil(n)\ *n* : a kiln or furnace for reducing limestone or shells to lime by burning

[1]lime-light \-,līt\ *n* 1 a : a stage lighting instrument producing illumination by means of an oxyhydrogen flame directed on a cylinder of lime and usu. equipped with a lens to concentrate the light in a beam b : the white light produced by such an instrument c *Brit* : SPOTLIGHT 2 : the center of public attention

[2]limelight *vt* : to center attention on : SPOTLIGHT

li-men \'lī-mən\ *n* [L *limin-, limen* — more at LIMB] : THRESHOLD 3a

limerick \'lim-(ə-)rik\ *n* [*Limerick*, Ireland] : a light or humorous verse form of 5 chiefly anapestic verses of which lines 1, 2, and 5 are of 3 feet and lines 3 and 4 are of 2 feet with a rhyme scheme of *aabba*

lime-stone \'līm-,stōn\ *n* : a rock that is formed chiefly by accumulation of organic remains (as shells or coral), consists mainly of calcium carbonate, is extensively used in building, and yields lime when burned

lime sulfur *n* : a fungicide and insecticide that contains calcium polysulfides and is usu. obtained by boiling sulfur with lime and water

lime-twig \'līm-,twig\ *n* 1 : a twig covered with birdlime to catch birds 2 : SNARE

lime-wa-ter \-,wȯt-ər, -,wät-\ *n* 1 : an alkaline water solution of calcium hydroxide used as an antacid 2 : natural water containing calcium carbonate or calcium sulfate in solution

limey \'lī-mē\ *n, pl* **limeys** *often cap* [*lime*-juicer + *-y*] 1 *slang* : a British sailor 2 *slang* : ENGLISHMAN

li-mic-o-line \lī-'mik-ə-,līn, -lən\ *adj* [deriv. of L *limus* mud + *colere* to inhabit; akin to L *linere* to smear — more at LIME, WHEEL] : inhabiting the shore region

lim-i-nal \'lim-ən-ᵊl\ *adj* [L *limin-, limen* threshold] 1 : of or relating to a sensory threshold 2 : barely perceptible

[1]limit \'lim-ət\ *n* [ME, fr. MF *limite*, fr. L *limit-, limes* boundary — more at LIMB] 1 a : a geographical or political boundary b *pl* : the place enclosed within a boundary : BOUNDS 2 a : something that bounds, restrains, or confines b : the utmost extent 3 : LIMITATION 4 : a determining feature or differentia in logic 5 : a prescribed maximum or minimum amount, quantity, or number: as a : the maximum quantity of game or fish that may be taken legally in a specified period b : a maximum established for a gambling bet, raise, or payoff 6 a : a number whose numerical difference from a mathematical function is arbitrarily small for all values of the independent variables that are sufficiently close to but not equal to given prescribed numbers or that are sufficiently large positively or negatively b : a number that for an infinite sequence of numbers is such that ultimately each of the remaining terms of the sequence differs from this number by less than any given amount 7 : something that is exasperating or intolerable — **limit-less** \-ləs\ *adj* — **limit-less-ly** *adv* — **limit-less-ness** *n*

[2]limit *vt* 1 : to assign certain limits to : PRESCRIBE ⟨reserved the right to ~ use of the land⟩ 2 a : to restrict to set bounds or limits ⟨the specialist can no longer ~ himself to his specialty⟩ b : to curtail or reduce in quantity or extent ⟨we must ~ the power of aggressors⟩ — **limit-able** \-ət-ə-bəl\ *adj* — **limit-er** *n*

syn LIMIT, RESTRICT, CIRCUMSCRIBE, CONFINE *shared meaning element* : to set bounds for. LIMIT implies setting a point or line (as in time, speed, space, or capacity) beyond which something cannot or is not permitted to go ⟨*limit* the working day to seven hours⟩ or it can imply bounds inherent in a situation or in the nature of something ⟨poor soil *limits* their crops⟩ RESTRICT usually connotes a narrowing or tightening or restraining within or as if within an encircling boundary ⟨*restrict* the powers of the president⟩ CIRCUMSCRIBE stresses a restricting in every direction and by clearly marked limits ⟨well-considered laws, *circumscribed* by a written constitution —V. L. Parrington⟩ CONFINE usually emphasizes bounds that cannot or must not be passed and often suggests severe restraint and the resulting cramping, fettering, or hampering ⟨*confined* to the house by illness⟩ ⟨now I am cabined, cribbed, *confined*, bound in to saucy doubts and fears —Shak.⟩ ***ant*** widen

limi-tary \'lim-ə-,ter-ē\ *adj* 1 *archaic* : subject to limits 2 a *archaic* : of or relating to a boundary b : LIMITING, ENCLOSING

limi-ta-tion \,lim-ə-'tā-shən\ *n* 1 : an act or instance of limiting 2 : the quality or state of being limited 3 : something that limits : RESTRAINT 4 : a certain period limited by statute after which actions, suits, or prosecutions cannot be brought in the courts — **limi-ta-tion-al** \-shnəl, -shən-ᵊl\ *adj*

limi-ta-tive \'lim-ə-,tāt-iv\ *adj* : serving to limit or restrict : LIMITING, RESTRICTIVE

limited *adj* 1 a : confined within limits : RESTRICTED ⟨~ success⟩ b *of a train* (1) : having a limited number of cars and making a limited number of stops (2) : offering superior and faster service and transportation 2 : characterized by enforceable limitations prescribed (as by a constitution) upon the scope or exercise of powers ⟨a ~ monarchy⟩ 3 : lacking breadth and originality ⟨a bit ~; a bit thick in the head —Virginia Woolf⟩ — **limit-ed-ly** *adv* — **limit-ed-ness** *n*

limited-access highway *n* : EXPRESSWAY

limited edition *n* : an edition of a publication limited to a specified number of copies and usu. printed in a special format

limited liability *n* : liability (as of a stockholder or shipowner) limited by statute or treaty

limited war *n* : a war whose objective is less than the total defeat of the enemy

limit-ing *adj* 1 : functioning as a limit : RESTRICTIVE ⟨~ factors⟩ 2 : serving to specify the application of the modified noun ⟨*this* in "this book" is a ~ word⟩

limit point *n* : a point that is related to a set of points in such a way that every neighborhood of the point no matter how small contains another point belonging to the set — called also *point of accumulation*

limi-trophe \'lim-ə-,trōf, -,trof\ *adj* [F] : situated on a border or frontier : ADJACENT

limmer \'lim-ər\ *n* [ME (Sc)] 1 *chiefly Scot* : SCOUNDREL 2 *chiefly Scot* : PROSTITUTE

limn \'lim\ *vt* **limned; limn-ing** \lim-(n)iŋ\ [ME *luminen, limnen* to illuminate (a manuscript), fr. MF *enluminer*, fr. L *illuminare* to illuminate] 1 : to draw or paint on a surface 2 : to outline in clear sharp detail : DELINEATE ⟨sees the tanker ~*ed* in her periscope sights —E. L. Beach⟩ 3 : DESCRIBE — **limn-er** \'lim-(n)ər\ *n*

limnetic \lim-'net-ik\ *adj* [ISV, fr. Gk *limnē* pool, marshy lake; akin to L *limen* threshold — more at LIMB] : of, relating to, or inhabiting the open water of a body of fresh water ⟨~ environment⟩

limnic \'lim-nik\ *adj* : LIMNETIC

limnology \lim-'näl-ə-jē\ *n* [Gk *limnē* + ISV *-logy*] : the scientific study of physical, chemical, meteorological, and biological conditions in fresh waters — **limno-log-i-cal** \,lim-nə-'läj-i-kəl\ *adj* — **limno-log-i-cal-ly** \-i-k(ə-)lē\ *adv* — **limnol-o-gist** \lim-'näl-ə-jəst\ *n*

limo \'lim-(,)ō\ *n, pl* **limos** : LIMOUSINE

limonene \'lim-ə-,nēn\ *n* ch bc [ISV, fr. F *limon* lemon] : a widely distributed terpene hydrocarbon $C_{10}H_{16}$ that occurs in essential oils (as of oranges or lemons) and has a lemon odor

limonite \'lī-mə-,nīt\ *n* [G *limonit*, fr. Gk *leimōn* meadow — more at LIMB] : a native hydrous ferric oxide of variable composition that is a major ore of iron — **limo-nit-ic** \,lī-mə-'nit-ik\ *adj*

limousine \'lim-ə-,zēn, ,lim-ə-'\ *n* [F, lit., cloak, fr. *Limousin*, France] 1 : a large luxurious often chauffeur-driven sedan that sometimes has a glass partition separating the driver's seat from the passenger compartment 2 : a small bus with doors along the side like those of a sedan ⟨an airport ~⟩

[1]limp \'limp\ *vi* [prob. fr. ME *lympen* to fall short; akin to OE *limpan* to happen, L *limbus* border, *labi* to slide — more at SLEEP] 1 a : to walk lamely; *esp* : to walk favoring one leg b : to go unsteadily : FALTER 2 : to proceed slowly or with difficulty ⟨commerce ~*ed* toward a standstill —*Time*⟩ — **limp-er** *n*

[2]limp *n* : a limping movement or gait

[3]limp *adj* [akin to **[1]limp**] 1 a : lacking or seeming to lack firmness and body and consequently drooping or shapeless ⟨~ curtains⟩ ⟨her hair hung ~ about her shoulders⟩ b : not stiff or rigid ⟨~

with thin cloth or paper bearing a device for public display (as for advertisement) and lighted from within

trans-par-ent \-ənt\ *adj* [ME, fr. ML *transparent-, transparens*, prp. of *transparēre* to show through, fr. L *trans-* + *parēre* to show oneself — more at APPEAR] **1 a** (1) : having the property of transmitting light without appreciable scattering so that bodies lying beyond are entirely visible : PELLUCID (2) : pervious to a specified form of radiation (as X rays or ultraviolet light) **b** : fine or sheer enough to be seen through : DIAPHANOUS **2 a** : free from pretense or deceit : FRANK **b** : easily detected or seen through : OBVIOUS **c** : readily understood : CLEAR — **trans-par-ent-ly** *adv* — **trans-par-ent-ness** *n*

trans-par-ent-ize \-ən-ˌtīz\ *vt* **-ized; -iz-ing** : to make transparent or more nearly transparent ⟨~ tracing paper⟩

trans-per-son-al \(ˌ)tran(t)s-ˈpərs-nəl, -ən-ᵊl\ *adj* : extending or going beyond the personal or individual

tran-spic-u-ous \tran(t)s-ˈpik-yə-wəs\ *adj* [NL *transpicuus*, fr. L *transpicere* to look through, fr. *trans-* + *specere* to look, see — more at SPY] : clearly seen through or understood

trans-pierce \tran(t)s-ˈpi(ə)rs\ *vt* [MF *transpercer*, fr. OF, fr. *trans-* (fr. L) + *percer* to pierce] : to pierce through : PENETRATE

tran-spi-ra-tion \tran(t)s-pə-ˈrā-shən\ *n* : the act or process or an instance of transpiring; *esp* : the passage of watery vapor from a living body through a membrane or pores

tran-spire \tran(t)s-ˈpī(ə)r\ *vb* **trans-pired; trans-pir-ing** [MF *transpirer*, fr. L *trans-* + *spirare* to breathe — more at SPIRIT] *vt* : to pass off or give passage to (a fluid) through pores or interstices; *esp* : to excrete (as water) in the form of a vapor through a living membrane (as the skin) ~ *vi* **1** : to give off vaporous material; *specif* : to give off or exude watery vapor esp. from the surfaces of leaves **2** : to pass in the form of a vapor from a living body **3 a** : to become known or apparent : DEVELOP **b** : to be revealed : come to light **4** : to come to pass : OCCUR ***syn*** see HAPPEN

trans-pla-cen-tal \ˌtran(t)s-plə-ˈsent-ᵊl\ *adj* [ISV] : passing through or occurring by way of the placenta ⟨~ immunization⟩ — **trans-pla-cen-tal-ly** \-ᵊl-ē\ *adv*

¹trans-plant \tran(t)s-ˈplant\ *vb* [ME *transplaunten*, fr. LL *transplantare*, fr. L *trans-* + *plantare* to plant] *vt* **1** : to lift and reset (a plant) in another soil or situation **2** : to remove from one place and settle or introduce elsewhere : TRANSPORT **3** : to transfer (an organ or tissue) from one part or individual to another ~ *vi* : to admit of being transplanted — **trans-plant-abil-i-ty** \ˌtran(t)s-ˌplant-ə-ˈbil-ət-ē\ *n* — **trans-plant-able** \tran(t)s-ˈplant-ə-bəl\ *adj* — **trans-plan-ta-tion** \ˌtran(t)s-ˌplan-ˈtā-shən\ *n* — **trans-plant-er** \tran(t)s-ˈplant-ər\ *n*

²trans-plant \ˈtran(t)s-ˌplant\ *n* **1** : the act or process of transplanting **2** : something transplanted

trans-po-lar \(ˌ)tran(t)s-ˈpō-lər\ *adj* : crossing or extending across either of the polar regions

tran-spon-der \tran(t)s-ˈpän-dər\ *n* [*trans*mitter + re*sponder*] : a radio or radar set that upon receiving a designated signal emits a radio signal of its own

trans-pon-tine \tran(t)s-ˈpän-ˌtīn\ *adj* [*trans-* + L *pont-, pons* bridge — more at FIND] **1** : situated on the farther side of a bridge **2** : resembling or characteristic of melodramas once popular in the theaters of London south of the Thames

¹trans-port \tran(t)s-ˈpō(ə)rt, -ˈpȯ(ə)rt, ˈtran(t)s-ˌ\ *vt* [ME *transporten*, fr. MF or L; MF *transporter*, fr. L *transportare*, fr. *trans-* + *portare* to carry — more at FARE] **1** : to transfer or convey from one place to another ⟨mechanisms of ~*ing* ions across a living membrane⟩ **2** : to carry away with strong and often intensely pleasant emotion **3** : to send to a penal colony overseas — **trans-port-abil-i-ty** \(ˌ)tran(t)s-ˌpōrt-ə-ˈbil-ət-ē, -ˌpȯrt-\ *n* — **trans-port-able** \tran(t)s-ˈpōrt-ə-bəl, -ˈpȯrt-\ *adj*

syn **1** see CARRY
2 TRANSPORT, RAVISH, ENRAPTURE, ENTRANCE *shared meaning element* : to carry away by strong and usu. pleasant emotion
3 see BANISH

²trans-port \ˈtran(t)s-ˌpō(ə)rt, -ˌpȯ(ə)rt\ *n* **1** : an act or process of transporting : TRANSPORTATION **2** : strong and often intensely pleasurable emotion ⟨~s of joy⟩ **3 a** : a ship for carrying soldiers or military equipment **b** : a vehicle (as a truck) used to transport persons or goods **c** : a system of public conveyance **4** : a transported convict **5** : a mechanism for moving tape and esp. magnetic tape past a sensing or recording head ***syn*** see ECSTASY

trans-por-ta-tion \ˌtran(t)s-pər-ˈtā-shən\ *n* **1** : an act, process, or instance of transporting or being transported **2** : banishment to a penal colony **3 a** : means of conveyance or travel from one place to another **b** : public conveyance of passengers or goods esp. as a commercial enterprise — **trans-por-ta-tion-al** \-shnəl, -shən-ᵊl\ *adj*

trans-port-er \tran(t)s-ˈpōrt-ər, -ˈpȯrt-, ˈtran(t)s-ˌ\ *n* : one that transports; *esp* : a vehicle for transporting large or heavy loads

¹trans-pose \tran(t)s-ˈpōz\ *vt* **trans-posed; trans-pos-ing** [ME *transposen*, fr. MF *transposer*, fr. L *transponere* (perf. indic. *transposui*) to change the position of, fr. *trans-* + *ponere* to put, place — more at POSITION] **1** : to change in form or nature : TRANSFORM **2** : to render into another language, style, or manner of expression : TRANSLATE **3** : to transfer from one place or period to another : SHIFT **4** : to change the relative place or normal order of : alter the sequence of ⟨~ letters to change the spelling⟩ **5** : to write or perform (a musical composition) in a different key **6** : to bring (a term) from one side of an algebraic equation to the other with change of sign ***syn*** see REVERSE — **trans-pos-able** \-ˈpō-zə-bəl\ *adj*

²trans-pose \ˈtran(t)s-ˌpōz\ *n* : a matrix formed by interchanging the rows and columns of a given matrix

trans-po-si-tion \ˌtran(t)s-pə-ˈzish-ən\ *n* [ML *transposition-, transpositio*, fr. L *transpositus*, pp. of *transponere* to transpose] **1** : an act, process, or instance of transposing or being transposed **2 a** : the transfer of any term of an equation from one side over to the other side with a corresponding change of the sign **b** : a mathematical permutation or interchange of two letters or symbols — **trans-po-si-tion-al** \-ˈzish-nəl, -ən-ᵊl\ *adj*

transposition cipher *n* : a cipher in which the letters of the plaintext are systematically rearranged into another sequence — compare SUBSTITUTION CIPHER

trans-sex-u-al \(ˌ)tran(t)s-ˈseksh(-ə)-wəl, -ˈsek-shəl\ *n* : a person genetically of one sex with a psychological urge to belong to the opposite sex that may be carried to the point of undergoing surgery to modify the sex organs to mimic the opposite sex — **trans-sex-u-al-ism** \-wə-ˌliz-əm, -shə-ˌliz-\ *n*

trans-shape \tran(ch)-ˈshāp, tran(t)s-\ *vt, archaic* : to change into another shape : TRANSFORM

trans-ship \tran(ch)-ˈship, tran(t)s-\ *vt* : to transfer for further transportation from one ship or conveyance to another ~ *vi* : to change from one ship or conveyance to another — **trans-ship-ment** \-mənt\ *n*

trans-tho-rac-ic \ˌtran(t)s-thə-ˈras-ik\ *adj* : done or made by way of the thoracic cavity — **trans-tho-rac-i-cal-ly** \-i-k(ə-)lē\ *adv*

tran-sub-stan-tial \ˌtran(t)s-əb-ˈstan-chəl\ *adj* : changed or capable of being changed from one substance to another

tran-sub-stan-ti-ate \ˌtran(t)s-əb-ˈstan-chē-ˌāt\ *vb* **-at-ed; -at-ing** [ML *transubstantiatus*, pp. of *transubstantiare*, fr. L *trans-* + *substantia* substance] *vt* **1** : to change into another substance : TRANSMUTE : to to effect transubstantiation in (sacramental bread and wine) ~ *vi* : to undergo transubstantiation

tran-sub-stan-ti-a-tion \ˌ-ˌstan-chē-ˈā-shən\ *n* **1** : an act or instance of transubstantiating or being transubstantiated **2** : the miraculous change by which according to Roman Catholic and Eastern Orthodox dogma the eucharistic elements at their consecration become the body and blood of Christ while keeping only the appearances of bread and wine

tran-su-date \tran(t)s-ˈ(y)üd-ət, tranz-, -ˌāt; ˈtran(t)s-(y)ü-ˌdāt, ˈtranz-\ *n* : a product of transudation

tran-su-da-tion \ˌtran(t)s-(y)ü-ˈdā-shən, ˌtranz-\ *n* **1** : the act or process of transuding or being transuded **2** : TRANSUDATE

tran-sude \tran(t)s-ˈ(y)üd, tranz-\ *vb* **tran-sud-ed; tran-sud-ing** [NL *transudare*, fr. L *trans-* + *sudare* to sweat — more at SWEAT] *vi* : to pass through a membrane or permeable substance : EXUDE ~ *vt* : to permit passage of

trans-ura-nic \ˌtran-shə-ˈran-ik, -ˈrā-nik, ˌtran-zhə-, ˌtran(t)s-yü-, ˌtranz-yü-\ *n* : a transuranium element

trans-ura-ni-um \-ˈrā-nē-əm\ *or* **trans-ura-nic** \-ˈran-ik, -ˈrā-nik\ *adj* : of, relating to, or being an element with an atomic number greater than that of uranium

trans-val-u-ate \(ˌ)tran(t)s-ˈval-yə-ˌwāt, (ˌ)tranz-\ *vt* **-at-ed; -at-ing** [back-formation fr. *transvaluation*] : TRANSVALUE

trans-val-u-a-tion \ˌtran(t)s-ˌval-yə-ˈwā-shən, ˌtranz-\ *n* : the act or process of transvaluing

trans-val-ue \(ˌ)tran(t)s-ˈval-(ˌ)yü, (ˌ)tranz-, -ˈval-yə(-w)\ *vt* **-val-ued; -valu-ing** : to reevaluate esp. on a basis that repudiates accepted standards

¹trans-ver-sal \tran(t)s-ˈvər-səl, tranz-\ *adj* : TRANSVERSE ⟨~ line⟩

²transversal *n* : a line that intersects a system of lines

¹trans-verse \tran(t)s-ˈvərs, tranz-, ˈtran(t)s-ˌ, ˈtranz-ˌ\ *adj* [L *transversus*, fr. pp. of *transvertere* to turn across, fr. *trans-* + *vertere* to turn — more at WORTH] **1** : lying or being across : set crosswise **2** : made at right angles to the anterior-posterior axis of the body ⟨a ~ section⟩ — **trans-verse-ly** *adv*

²trans-verse \ˈtran(t)s-ˌvərs, ˈtranz-\ *n* : something (as a piece, section, or part) that is transverse

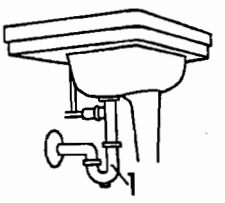
transverse colon *n* : the middle portion of the colon that extends across the abdominal cavity

transverse process *n* : a lateral process of a vertebra — see VERTEBRA illustration

transverse wave *n* : a wave in which the vibrating element moves in a direction perpendicular to the direction of advance of the wave

trans-ves-tism \tran(t)s-ˈves-ˌtiz-əm, tranz-\ *n* [G *transvestismus*, fr. L *trans-* + *vestire* to clothe — more at VEST] : adoption of the dress and often the behavior of the opposite sex — **trans-ves-tite** \-ˌtīt\ *adj or n*

¹trap \ˈtrap\ *n* [ME, fr. OE *treppe* & OF *trape* (of Gmc origin); akin to MD *trappe* trap, stair, OE *treppan* to tread, Skt *drvati* he runs] **1** : a device for taking game or other animals; *esp* : one that holds by springing shut suddenly **2 a** : something by which one is caught or stopped unawares **b** : a football play in which a defensive player is allowed to cross the line of scrimmage and then is blocked from the side while the ballcarrier advances through the spot vacated by the defensive player **3 a** : a device for hurling clay pigeons into the air **b** : SAND TRAP **c** : a piece of leather or section of interwoven leather straps between the thumb and forefinger of a baseball glove that forms an extension of the pocket **4** *slang* : MOUTH **5** : a light usu. one-horse carriage with springs **6** : any of various devices for preventing passage of something often while allowing other matter to proceed; *esp* : a device for drains or sewers consisting of a bend or partitioned chamber in which the liquid forms a seal to prevent the passage of sewer gas **7** *pl* : a group of percussion instruments (as a bass drum, snare drums, and cymbals) used esp. in a dance or jazz band **8** *pl* [*speed trap*] : a



1, trap 6

[1]wise-crack \'wīz-ˌkrak\ *n* : a sophisticated or knowing witticism *syn* see JEST
[2]wisecrack *vi* : to make a wisecrack — **wise-crack-er** *n*
wise guy \'wīz-ˌgī\ *n* : a cocky conceited fellow : KNOW-IT-ALL
wise man *n* **1** : a man of unusual learning, judgment, or insight : SAGE **2** : a man versed in esoteric lore (as of magic or astrology); *specif* : MAGUS 2
wi-sen-hei-mer \'wīz-ᵊn-ˌhī-mər\ *n* [*wise* + G *-enheimer* (as in G family names such as *Guggenheimer, Oppenheimer*)] : one who has the air of knowing all about something : WISEACRE
wi-sent \'vē-ˌzent\ *n* [G, fr. OHG *wisunt* — more at BISON] : a European bison (*Bison bonasus*) — called also *aurochs*
wise-wom-an \'wīz-ˌwům-ən\ *n* **1** : a woman versed in charms, conjuring, or fortune-telling **2** : MIDWIFE
[1]wish \'wish\ *vb* [ME *wisshen*, fr. OE *wȳscan;* akin to OHG *wunsken* to wish, L *venus* love, charm — more at WIN] *vt* **1** : to have a desire for (as something unattainable) : WANT ⟨~*ed* he could live his life over⟩ **2** : to give expression to as a wish : BID ⟨~ him good night⟩ **3 a** : to give form to (a wish) **b** : to express a wish for **c** : to request in the form of a wish : ORDER **4** : to confer (something unwanted) upon someone : FOIST ~ *vi* **1** : to have a desire : WANT **2** : to make a wish *syn* see DESIRE — **wish-er** *n*
[2]wish *n* **1 a** : an act or instance of wishing or desire : WANT **b** : an object of desire : GOAL **2 a** : an expressed will or desire : MANDATE **b** : a request or command couched as a wish **3** : an invocation of good or evil fortune on someone
wisha \'wish-ə\ *interj* [IrGael *ō* oh + *muise* indeed] *chiefly Irish* — used as an intensive or to express surprise
wish-bone \'wish-ˌbōn\ *n* [fr. the superstition that when two persons pull it apart the one getting the longer fragment will have his wish granted] **1** : a furcula in front of the breastbone in a bird consisting chiefly of the two clavicles fused at their median or lower end **2** : a variation of the T formation in which the halfbacks line up farther from the line of scrimmage than the fullback does
wish-ful \'wish-fəl\ *adj* **1 a** : expressive of a wish : HOPEFUL **b** : having a wish : DESIROUS **2** : according with wishes rather than reality — **wish-ful-ly** \-fə-lē\ *adv* — **wish-ful-ness** *n*
wish fulfillment *n* : the gratification of a desire esp. as gained symbolically (as in dreams, daydreams, or neurotic symptoms)
wishful thinking *n* **1** : the attribution of reality to what one wishes to be true and the tenuous justification of what one wants to believe **2** : AUTISM
wish-ing *adj* **1** *archaic* : WISHFUL **2** : regarded as having the power to grant wishes ⟨threw a coin in the ~ well⟩
wish-wash \'wish-ˌwȯsh, -ˌwäsh\ *n* [redupl. of *[2]wash*] **1** : a weak drink **2** : insipid talk or writing
wishy-washy \'wish-ē-ˌwȯsh-ē, -ˌwäsb-\ *adj* [redupl. of *washy*] **1** : lacking in strength or flavor : WEAK **2** : lacking in character or determination : INEFFECTUAL
[1]wisp \'wisp\ *n* [ME] **1** : a small handful (as of hay or straw) **2 a** : a thin strip or fragment **b** : a thready streak ⟨a ~ of smoke⟩ **c** : something frail, slight, or fleeting ⟨a ~ of a girl⟩ ⟨a ~ of a smile⟩ **3** : WILL-O'-THE-WISP — **wispy** \'wis-pē\ *adj*
[2]wisp *vt* **1** : to roll into a wisp **2 a** : to make wisps of ⟨a cigarette ~*ing* smoke at the corner of his mouth —Raymond Chandler⟩ **b** : to cover with wisps ⟨the sky all ~*ed* with mist —W. F. Wray⟩ ~ *vi* : to emerge or drift in wisps ⟨her hair began to ~ into her eyes —Mary Manning⟩
wisp-ish \'wis-pish\ *adj* : resembling a wisp : INSUBSTANTIAL
wist \'wist\ *vt* [alter. of *wis*] *archaic* : KNOW
wis-tar-ia \wis-'tir-ē-ə *also* -'ter-\ *n* [NL, alter. of *Wisteria*] : WISTERIA
wis-te-ria \-'tir-ē-ə\ *n* [NL, genus name, fr. Caspar *Wistar* †1818 Am physician] : any of a genus (*Wisteria*) of chiefly Asiatic mostly woody leguminous vines having pinnately-compound leaves and showy blue, white, purple, or rose pealike flowers in long racemes and including several grown as ornamentals
wist-ful \'wist-fəl\ *adj* [blend of *wishful* and obs. E *wistly* (intently)] **1** : full of unfulfilled longing or desire : YEARNING **2** : musingly sad : PENSIVE — **wist-ful-ly** \-fə-lē\ *adv* — **wist-ful-ness** *n*
[1]wit \'wit\ *vb* **wist** \'wist\; **wit-ting**; *pres 1st & 3d sing* **wot** \'wät\ [ME *witen* (1st & 3d sing. pres. *wot*, past *wiste*), fr. OE *witan* (1st & 3d sing. pres. *wāt*, past *wisse, wiste*); akin to OHG *wizzan* to know, L *vidēre* to see, Gk *eidenai* to know, *idein* to see] **1** *archaic* : KNOW **2** *archaic* : to come to know : LEARN
[2]wit *n* [ME, fr. OE; akin to OHG *wizzi* knowledge, OE *witan* to know] **1 a** : MIND, MEMORY **b** : reasoning power : INTELLIGENCE **2 a** : SENSE 2a — usu. used in pl. ⟨alone and warming his five ~s, the white owl in the belfry sits —Alfred Tennyson⟩ **b** (1) : mental soundness : SANITY — usu. used in pl. (2) : mental capability and resourcefulness : INGENUITY **3 a** : astuteness of perception or judgment : ACUMEN **b** : the ability to relate seemingly disparate things so as to illuminate or amuse **c** (1) : a talent for banter or persiflage (2) : a witty utterance or exchange **4 a** : a person of superior intellect : THINKER **b** : an imaginatively perceptive and articulate individual esp. skilled in banter or persiflage
syn WIT, HUMOR, IRONY, SARCASM, SATIRE, REPARTEE *shared meaning element* : a mode of expression intended to arouse amused interest or evoke attention and laughter or a quality of mind that predisposes to such expression. WIT suggests the power to evoke laughing attention by remarks showing verbal felicity or ingenuity and swift perception, especially of the incongruous ⟨true *wit* is nature to advantage dressed, what oft was thought, but ne'er so well expressed —Alexander Pope⟩ HUMOR implies an ability to perceive and effectively express the ludicrous, the comical, or the absurd, especially in human life ⟨the modern sense of *humor* is the quiet enjoyment and implicit expression of the fun of things — Louis Cazamian⟩ IRONY applies to a manner of presentation in which an intended meaning is subtly emphasized by appropriate expression of its opposite ⟨*irony* properly suggests the opposite of what is explicitly stated, by means of peripheral clues — tone of voice, accompanying gestures, stylistic exaggeration . . . thus, for "Brutus is an honorable man" we understand "Brutus is a traitor" —Jacob Brackman⟩ SARCASM applies to savagely humorous expression, frequently in the form of irony, intended to cut and wound ⟨the arrows of *sarcasm* are barbed with contempt —Washington Gladden⟩ SATIRE applies primarily to writing that holds up vices or follies to ridicule and reprobation often by use of irony or caricature ⟨his dry wit and his easy, good-natured *satire* on the follies of the day —Eleanor M. Sickels⟩ REPARTEE applies to the power or art of responding quickly, smoothly, pointedly, and wittily or to an interchange of such response ⟨as for *repartee* . . . as it is the very soul of conversation, so it is the greatest grace of comedy —John Dryden⟩
— **at one's wit's end** *or* **at one's wits' end** : at a loss for a means of solving a problem
wi-tan \'wi-ˌtän\ *n pl* [OE, pl. of *wita* sage, adviser; akin to OHG *wizzo* sage, OE *witan* to know] : members of the witenagemot
[1]witch \'wich\ *n* [ME *wicche*, fr. OE *wicca*, masc., wizard & *wicce*, fem., witch; akin to MHG *wicken* to bewitch, OE *wigle* divination, OHG *wih* holy — more at VICTIM] **1** : one that is credited with usu. malignant supernatural powers; *esp* : a woman practicing usu. black witchcraft often with the aid of a devil or familiar : SORCERESS — compare WARLOCK **2** : an ugly old woman : HAG **3** : a charming or alluring girl or woman — **witchy** \'wich-ē\ *adj*
[2]witch *vt* **1** : to affect injuriously with witchcraft **2** *archaic* : to influence or beguile with allure or charm ~ *vi* : DOWSE
witch-craft \'wich-ˌkraft\ *n* **1 a** : the use of sorcery or magic **b** : communication with the devil or with a familiar **2** : an irresistible influence or fascination : ENCHANTMENT
witch doctor *n* : a professional worker of magic usu. in a primitive society who often works to cure sickness
witch-ery \'wich-(ə-)rē\ *n, pl* **-er-ies** **1 a** : the practice of witchcraft : SORCERY **b** : an act of witchcraft **2** : an irresistible fascination : CHARM
witches' brew *n* : a potent or fearsome mixture ⟨a *witches' brew* of untamed sex and brutality —Harrison Smith⟩
witch-es'-broom \'wich-əz-ˌbrüm, -ˌbrům\ *n* : an abnormal tufted growth of small branches on a tree or shrub caused esp. by fungi or viruses — called also *hexenbesen*
witches' Sabbath *n* : a midnight assembly of witches, devils, and sorcerers for the celebration of rites and orgies
witch-grass \'wich-ˌgras\ *n* [prob. alter. of *quitch* (*grass*)] **1** : QUACK GRASS **2** [*[1]witch*] : a No. American grass (*Panicum capillare*) with slender brushy panicles that is often a weed on cultivated land
witch ha-zel \'wich-ˌhā-zəl\ *n* [*witch* (a tree with pliant branches)] **1** : any of a genus (*Hamamelis* of the family Hamamelidaceae, the witch-hazel family) of shrubs with slender-petaled yellow flowers borne in late fall or early spring; *esp* : one (*H. virginiana*) of eastern No. America that blooms in the fall **2** : an alcoholic solution of a distillate of the bark of a witch hazel (*H. virginiana*) used as a soothing and mildly astringent lotion


witch hazel 1

witch-hunt \'wich-ˌhənt\ *n* **1** : the searching out and deliberate harassment of those (as political opponents) with unpopular views **2** : a searching out for persecution of persons accused of witchcraft — **witch-hunt-er** *n* — **witch-hunt-ing** *n or adj*
[1]witch-ing \'wich-iŋ\ *n* : the practice of witchcraft : SORCERY
[2]witching *adj* : of, relating to, or suitable for sorcery or supernatural occurrences ⟨the very ~ time of night —Shak.⟩
witch-like \'wich-ˌlīk\ *adj* : resembling or befitting a witch
witch moth *n* : any of various noctuid moths (as of the genus *Erebus*)
witch of Agne-si \-ən-'yā-zē\ [Maria Gaetana *Agnesi* †1799 It mathematician; probably from its resemblance to the outline of a witch's hat] : a plane cubic curve that is symmetric about the y-axis, approaches the x-axis as an asymptote, and has the equation $x^2y = 4a^2(2a - y)$ — called also *witch*
witch-weed \'wich-ˌwēd\ *n* : any of a genus (*Striga* of the figwort family) of yellow-flowered Old World plants that are damaging root parasites of grasses (as sorghum and maize) and that include one (*S. asiatica*) which is an introduced pest in parts of the southeastern U.S.
[1]wite \'wit\ *n* [ME, fr. OE *wite* punishment; akin to OHG *wizi* punishment, OE *witan* to know] *chiefly Scot* : BLAME, RESPONSIBILITY
[2]wite *vt* **wit-ed; wit-ing** *chiefly Scot* : BLAME
wi-te-na-ge-mot *or* **wi-te-na-ge-mote** \'wit-ᵊn-ə-gə-ˌmōt\ *n* [OE *witena gemōt*, fr. *witena* (gen. pl. of *wita* sage, adviser) + *gemōt* gemot] : an Anglo-Saxon council made up of a varying number of nobles, prelates, and influential officials and convened from time to time to advise the king on administrative and judicial matters — compare WITAN
with \(')with, (')wəth, wəth\ *prep* [ME, against, from, with, fr. OE; akin to OE *wither* against, OHG *widar* against, back, Skt *vi* apart] **1 a** : in opposition to : AGAINST ⟨had a fight ~ his brother⟩ **b** : so as to be separated or detached from ⟨broke ~ his family⟩ **2** — used as a function word to indicate one to whom a usu. reciprocal communication is made ⟨talking ~ a friend⟩ **3 a** — used as a function word to indicate one that shares in an action, transaction, or arrangement ⟨works ~ his father⟩ **b** — used as a function word to indicate the object of attention, behavior, or feeling ⟨get tough ~ him⟩ ⟨angry ~ her⟩ **c** : in respect to : so far as concerns ⟨on friendly terms ~ all nations⟩ **d** — used to indicate the object of an adverbial expression of imperative force ⟨off ~ his head⟩ **e** : OVER, ON ⟨no longer has any influence ~ him⟩ **f** : in the performance, operation, or use of ⟨the trouble ~ this machine⟩ **4 a** — used as a function word to indicate the object of a statement of comparison or equality ⟨a dress identical ~ her hostess's⟩

b — used as a function word to express agreement or sympathy ⟨must conclude, ~ him, that the painting is a forgery⟩ **c** : on the side of : FOR ⟨if he's for lower taxes, I'm ~ him⟩ **d** : as well as ⟨can pitch ~ the best of them⟩ **5 a** — used as a function word to indicate combination, accompaniment, presence, or addition ⟨heat milk ~ honey⟩ ⟨went there ~ her⟩ ⟨his money, ~ his wife's, comes to a million⟩ **b** : inclusive of ⟨costs five dollars ~ the tax⟩ **6 a** : in the judgment or estimation of ⟨stood well ~ his classmates⟩ **b** : in or according to the experience or practice of ⟨~ many of us, our ideas seem to fall by the wayside —W. J. Reilly⟩ **7 a** — used as a function word to indicate the means, cause, agent, or instrumentality ⟨hit him ~ a rock⟩ ⟨pale ~ anger⟩ ⟨threatened ~ tuberculosis⟩ ⟨he amused the crowd ~ his antics⟩ **b** *archaic* : by the direct act of **8 a** — used as a function word to indicate manner of action ⟨ran ~ effort⟩ ⟨acknowledge your contribution ~ thanks⟩ **b** — used as a function word to indicate an attendant fact or circumstance ⟨stood there ~ his hat on⟩ **c** — used as a function word to indicate a result attendant on a specified action ⟨got off ~ a light sentence⟩ **9 a** (1) : in possession of : HAVING ⟨came ~ good news⟩ (2) : in the possession or care of ⟨left the money ~ his mother⟩ **b** : characterized or distinguished by ⟨a man ~ a sharp nose⟩ **10 a** — used as a function word to indicate a close association in time ⟨~ the outbreak of war they went home⟩ ⟨mellows ~ time⟩ **b** : in proportion to ⟨the pressure varies ~ the depth⟩ **11 a** : in spite of : NOTWITHSTANDING ⟨a really tip-top man, ~ all his wrongheadedness —H. J. Laski⟩ **b** : except for ⟨finds that, ~ one group of omissions and one important addition, they reflect that curriculum —Gilbert Highet⟩ **12** : in the direction of ⟨~ the wind⟩ ⟨~ the grain⟩ ***syn*** see BY

¹with-al \with-'ȯl, with-\ *adv* [ME, fr. *with* + *all, al* all] **1** : together with this : BESIDES ⟨a supporter of all constructive work and ~ an excellent businessman —A. W. Long⟩ **2** *archaic* : THEREWITH **3** : on the other hand : NEVERTHELESS

²withal *prep, archaic* : WITH — used postpositively with a relative or interrogative pronoun as object

with-draw \with-'drȯ, with-\ *vb* **-drew** \-'drü\; **-drawn** \-'drȯn\; **-draw-ing** \-'drȯ(-)iŋ\ [ME, fr. *with* from + *drawen* to draw] *vt* **1 a** : to take back or away : REMOVE ⟨pressure upon educational administrators to ~ academic credit —J. W. Scott⟩ **b** : to remove from use or cultivation **c** : to remove (money) from a place of deposit **d** : to turn away (as the eyes) from an object of attention ⟨*withdrew* his gaze⟩ **e** : to draw (as a curtain) back or aside **2 a** : to remove from consideration or set outside a group ⟨*withdrew* his name from the list of nominees⟩ ⟨*withdrew* his son from the school⟩ **b** (1) : to take back : RETRACT (2) : to recall or remove (a motion) under parliamentary procedure ~ *vi* **1 a** : to move back or away : RETIRE **b** : to draw back from a battlefield : RETREAT **2 a** : to remove oneself from participation **b** : to become socially or emotionally detached ⟨had *withdrawn* farther and farther into herself —Ethel Wilson⟩ **3** : to recall a motion under parliamentary procedure ***syn*** see GO — **with-draw-able** \-'drȯ-ə-bəl\ *adj*

with-draw-al \-'drȯ(ə-)l\ *n* **1 a** : retreat or retirement esp. into a more secluded or less exposed place or position **b** : an operation by which a military force disengages from the enemy **c** (1) : social or emotional detachment (2) : a pathological retreat from objective reality (as in some schizophrenic states) **2** : RETRACTION, REVOCATION ⟨threatened us with ~ of his consent⟩ **3** : the act of drawing someone or something back from or out of a place or position **4 a** : the act of taking back or away something that has been granted or possessed **b** : removal from a place of deposit or investment **c** : the discontinuance of administration or use of a drug

withdrawing room *n* : a room to retire to (as from a dining room); *esp* : DRAWING ROOM

with-drawn \with-'drȯn\ *adj* **1** : removed from immediate contact or easy approach : ISOLATED **2** : socially detached and unresponsive : INTROVERTED — **with-drawn-ness** \-'drȯn-nəs\ *n*

withe \'with, 'with, 'with\ *n* [ME, fr. OE *withthe*; akin to OE *withig* withy] : a slender flexible branch or twig; *esp* : one used as a band or line

with-er \'with-ər\ *vb* **with-ered; with-er-ing** \-(ə-)riŋ\ [ME *widren*; prob. akin to ME *weder* weather] *vi* **1** : to become dry and sapless; *esp* : to shrivel from or as if from loss of bodily moisture **2** : to lose vitality, force, or freshness ~ *vt* **1** : to cause to wither **2** : to make speechless or incapable of action : STUN ⟨~*ed* him with a look —Dorothy Sayers⟩

with-er-ing *adj* : acting or serving to cut down or destroy : DEVASTATING ⟨a ~ fire from the enemy⟩ — **with-er-ing-ly** \-(ə-)riŋ-lē\ *adv*

with-er-ite \'with-ə-ˌrīt\ *n* [G *witherit*, irreg. fr. William *Withering* †1799 E physician] : a mineral $BaCO_3$ consisting of a carbonate of barium in the form of white or gray twin crystals or columnar or granular masses

withe rod *n* : either of two No. American viburnums (*Viburnum cassinoides* and *V. nudum*) with tough slender shoots

with-ers \'with-ərz\ *n pl* [prob. fr. obs. E *wither-* (against), fr. ME, fr. OE, fr. *wither* against; fr. the withers being the parts which resist the pull in drawing a load — more at WITH] **1** : the ridge between the shoulder bones of a horse — see HORSE illustration **2** : a part corresponding to the withers in a quadruped other than a horse

with-er-shins \'with-ər-shənz\ *var of* WIDDERSHINS

with-hold \with-'hōld, with-\ *vb* **-held** \-'held\; **-hold-ing** [ME *withholden*, fr. *with* from + *holden* to hold — more at WITH] *vt* **1** : to hold back from action : CHECK **2** *archaic* : to keep in custody **3** : to refrain from granting, giving, or allowing ⟨~ permission⟩ **4** : to deduct (withholding tax) from income ~ *vi* : FORBEAR, REFRAIN ***syn*** see KEEP — **with-hold-er** *n*

withholding tax *n* : a deduction (as from wages, fees, or dividends) levied at a source of income as advance payment on income tax

¹with-in \with-'in, with-\ *adv* [ME *withinne*, fr. OE *withinnan*, fr. *with* + *innan* inwardly, within, fr. *in*] **1** : in or into the interior : INSIDE **2** : in one's inner thought, disposition, or character : INWARDLY ⟨search ~ for a creative impulse — Kingman Brewster, Jr.⟩

²within *prep* **1** — used as a function word to indicate enclosure or containment **2** — used as a function word to indicate situation or circumstance in the limits or compass of: as **a** : before the end of ⟨gone ~ a week⟩ **b** (1) : not beyond the quantity, degree, or limitations of ⟨lives ~ his income⟩ (2) : in or into the scope or sphere of ⟨~ the jurisdiction of the state⟩ (3) : in or into the range of ⟨~ reach⟩ ⟨~ sight⟩ (4) — used as a function word to indicate a specified difference or margin ⟨came ~ two points of a perfect mark⟩ ⟨~ a mile of the town⟩ **3** : to the inside of : INTO

³within *n* : an inner place or area ⟨revolt from ~⟩

⁴within *adj* : being inside : ENCLOSED ⟨the ~ indictment⟩

with-in-doors \with-ˌin-'dō(ə)rz, with-, -'dȯ(ə)rz\ *adv* : INDOORS

with-it \'with-ət\ *adj* : attuned to a social or cultural vanguard : socially or culturally up-to-date ⟨the intelligent, disaffected, ~ young —Eliot Fremont-Smith⟩

¹with-out \with-'au̇t, with-\ *prep* [ME *withoute*, fr. OE *withūtan*, fr. *with* + *ūtan* outside, fr. *ūt* out] **1** : OUTSIDE **2** — used as a function word to indicate the absence or lack of something or someone ⟨fight ~ fear⟩ ⟨left ~ him⟩ ⟨looks ~ seeing⟩

²without *adv* **1** : on the outside : EXTERNALLY **2** : with something lacking or absent ⟨has learned to do ~⟩

³without *conj, chiefly dial* : UNLESS ⟨you don't know about me ~ you have read a book —Mark Twain⟩

⁴without *n* : an outer place or area ⟨came from ~⟩

with-out-doors \with-ˌau̇t-'dō(ə)rz, with-\ *adv* : OUTDOORS

with-stand \with-'stand, with-\ *vt* **-stood** \-'stu̇d\; **-stand-ing** [ME *withstanden*, fr. OE *withstandan*, fr. *with* against + *standan* to stand — more at WITH] **1 a** : to stand up against : oppose with firm determination; *esp* : to resist successfully **b** : to be proof against : resist the effect of ⟨~ the impact of a landing —*Current Biog.*⟩ **2** *archaic* : to stop or obstruct the course of ***syn*** see OPPOSE

¹withy \'with-ē\ *n, pl* **with-ies** [ME, fr. OE *withig*; akin to OHG *wida* willow, L *vitis* vine, *viēre* to plait — more at WIRE] **1** : WILLOW; *esp* : OSIER 1 **2** : a flexible slender twig or branch (as of osier) : WITHE

²withy \'with-ē, 'with-ē, 'wī-thē\ *adj* [*withe*] : flexibly tough

wit-less \'wit-ləs\ *adj* **1** : destitute of wit or understanding : FOOLISH **2** : mentally deranged : CRAZY ⟨drive one ~ with anxiety —William Styron⟩

wit-ling \-liŋ\ *n* : a person of little wit

wit-loof \'wit-ˌlōf, -ˌlu̇f\ *n* [D dial. *witloof* chicory, fr. D *wit* white + *loof* foliage] : CHICORY 1; *also* : ENDIVE 2

¹wit-ness \'wit-nəs\ *n* [ME *witnesse*, fr. OE *witnes* knowledge, testimony, witness, fr. ²*wit*] **1** : attestation of a fact or event : TESTIMONY **2** : one that gives evidence; *specif* : one who testifies in a cause or before a judicial tribunal **3** : one asked to be present at a transaction so as to be able to testify to its having taken place **4** : one who has personal knowledge of something **5 a** : something serving as evidence or proof : SIGN **b** : public affirmation by word or example of usu. religious faith or conviction ⟨the heroic ~ to divine life —*Pilot*⟩ **6** *cap* : a member of the Jehovah's Witnesses

²witness *vt* **1** : to testify to : ATTEST **2** : to act as legal witness of **3** : to furnish proof of : BETOKEN **4 a** : to have personal or direct cognizance of : see for oneself ⟨~*ed* the historic event⟩ **b** : to take note of ⟨our grammar — ~ our verb system — is a marvel of flexibility, variety, and exactitude —Charlton Laird⟩ **5** : to constitute the scene or time of ⟨structures . . . which this striking Dorset hilltop once ~*ed* —*Times Lit. Supp.*⟩ ~ *vi* **1** : to bear witness : TESTIFY **2** : to bear witness to one's religious convictions ⟨opportunity to ~ for Christ —W. F. Graham⟩

wit-ness-box \-ˌbäks\ *n, chiefly Brit* : an enclosure in which a witness sits or stands while testifying in court

witness stand *n* : a stand or an enclosure from which a witness gives evidence in a court

wit-ted \'wit-əd\ *adj* : having wit or understanding — usu. used in combination ⟨dull-*witted*⟩

wit-ti-cism \'wit-ə-ˌsiz-əm\ *n* [*witty* + *-cism* (as in *criticism*)] : a cleverly witty and often biting or ironic remark ***syn*** see JEST

¹wit-ting \'wit-³n, -iŋ\ *n* **1** *chiefly dial* : knowledge or awareness of something : COGNIZANCE **2** *chiefly dial* : information obtained or communicated : NEWS

²wit-ting \-iŋ\ *adj* **1** : cognizant or aware of something : CONSCIOUS **2** : done deliberately : INTENTIONAL — **wit-ting-ly** \-iŋ-lē\ *adv*

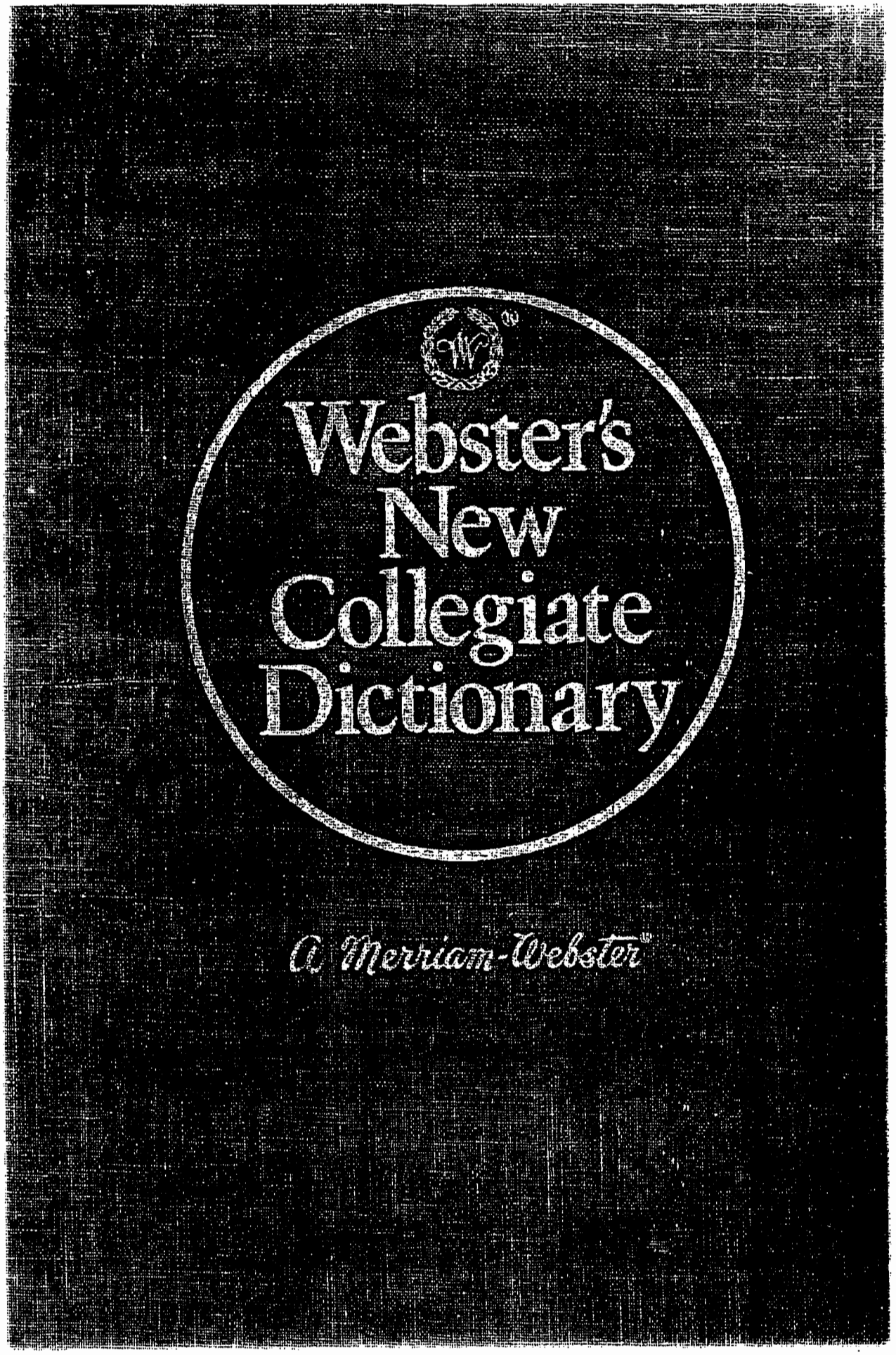
wit-tol \'wit-³l\ *n* [ME *wetewold*, fr. *weten, witen* to know + *-wold* (as in *cokewold* cuckold) — more at WIT] **1** *archaic* : a man who knows of his wife's infidelity and puts up with it **2** *archaic* : a witless person

wit-ty \'wit-ē\ *adj* **wit-ti-er; -est** **1** *archaic* : having good intellectual capacity : INTELLIGENT **2** : amusingly or ingeniously clever in conception or execution ⟨the costumes are sumptuous and ~ — Virgil Thomson⟩ ⟨the musical background is . . . often ~ — Wolcott Gibbs⟩ **3** : marked by or full of wit : smartly facetious or jocular **4** : quick or ready to see or express illuminating or amusing relationships or insights — **wit-ti-ly** \'wit-³l-ē\ *adv* — **wit-ti-ness** \'wit-ē-nəs\ *n*

syn WITTY, HUMOROUS, FACETIOUS, JOCULAR, JOCOSE *shared meaning element* : provoking or tending to provoke amusement or laughter

wive \'wīv\ *vb* **wived; wiv-ing** [ME *wiven*, fr. OE *wīfian*, fr. *wif* woman, wife] *vi* : to marry a woman ~ *vt* **1** : to marry to a wife **2** : to take for a wife

TAB 29


Webster's
New
Collegiate
Dictionary
A Merriam-Webster

New Collegiate Dictionary

A Merriam-Webster®

G. & C. MERRIAM COMPANY
Springfield, Massachusetts, U.S.A.

Library of Congress Cataloging in Publication Data
Main entry under title:

Webster's new collegiate dictionary.

Editions for 1898–1948 have title: Webster's collegiate dictionary.
Includes index.
1. English language—Dictionaries.
PE1628.W4M4 1981 423 80-25144
ISBN 0-87779-408-1
ISBN 0-87779-409-x (indexed)
ISBN 0-87779-410-3 (deluxe)

Made in the United States of America

504948 RMcN8281

Contents

Preface

Webster's New Collegiate Dictionary is a completely new volume in the Merriam-Webster series of dictionaries. It is a general dictionary edited for use in school or college, in the office, and in the home—in short, wherever information about English words is likely to be sought. The average user should rarely have occasion to look for information about the vocabulary of present-day English that is not available within these pages.

The first Merriam-Webster Collegiate appeared in 1898 and quickly won the esteem of student and general reader. A second edition was published in 1910, and subsequent editions came out in 1916, 1931, 1936, 1949, and 1963. This eighth in the series incorporates the best of the time-tested features of its predecessors and introduces new features designed to add to its usefulness. Its more than 1500 pages make it the most comprehensive Merriam-Webster Collegiate ever published.

The heart of Webster's New Collegiate Dictionary is the more than 1300 pages given over to the A–Z vocabulary. The information there set down derives not only from the 10,000,000 citations which were available to the editors of Webster's Third New International Dictionary and the 1963 Collegiate but also from the considerably more than 1,000,000 citations collected since the publication of these books. Thus each entry is based on a constantly updated file of actual English usage.

Those entries known to be trademarks or service marks are so labeled and are treated in accordance with a formula approved by the United States Trademark Association. No entry in this dictionary, however, should be regarded as affecting the validity of any trademark or service mark.

A noteworthy feature of the vocabulary section is the nearly 900 pictorial illustrations, many of which were drawn especially for this book. These illustrations were selected not simply for their decorative function but particularly for their value in clarifying definitions.

The front matter—those pages preceding the A–Z vocabulary—contains two important sections. The Explanatory Notes should be read by every user of the dictionary since a thorough understanding of the information contained in them will contribute markedly to the value of this book. And all users of the dictionary are urged to read the lucid essay on the English language which was written for this Collegiate by Professor W. Nelson Francis of Brown University.

The back matter—those pages following the A–Z vocabulary—contains several sections that dictionary users have long found helpful. These include more than five hundred Foreign Words and Phrases that occur frequently in English texts but that have not become part of the English vocabulary; several thousand proper names that are entered under the separate headings Biographical Names and Geographical Names; and a list of the Colleges and Universities of the United States and Canada. There is also a Handbook of Style in which various stylistic conventions (as of punctuation and capitalization) are concisely summarized.

Webster's New Collegiate Dictionary has been edited by the trained staff of the G. & C. Merriam Co. It is the result of a collaborative effort, and it would be invidious to single out particular editors for special mention. At the same time, it would be ungracious to observe the anonymity which is often the lot of the present-day lexicographer, and so a list of those who contributed substantially to the completion of this book is printed below.

Webster's New Collegiate Dictionary is the product of a company that has been publishing dictionaries for more than 125 years. It is offered to the user with the conviction that it will serve him well.

Henry Bosley Woolf
Editor in Chief

Editorial Staff

Editor in Chief

Henry Bosley Woolf

Senior Editors

Edward Artin • F. Stuart Crawford • E. Ward Gilman • Mairé Weir Kay • Roger W. Pease, Jr.

Associate Editors

Robert D. Copeland • Grace A. Kellogg • Hubert P. Kelsey • James G. Lowe • George M. Sears

Assistant Editors

William Parr Black • Kathryn K. Flynn • Dolores R. Harris • Laverne W. King • Kerry W. Metz • Trudy A. Perkins • James E. Shea, Jr. • Anne H. Soukhanov • Raymond R. Wilson

Editorial Assistants

Dwight H. Day, Jr. • Philip B. Dickinson • Kathleen M. Doherty • L. Aimee Garn • Peter B. Kent

Editorial Consultants

Charlotte A. Bridgman • Philip W. Cummings • Philip B. Gove

Librarian

Marion D. Ware

Departmental Secretary

Hazel O. Lord

Head of Typing Room

Evelyn G. Summers

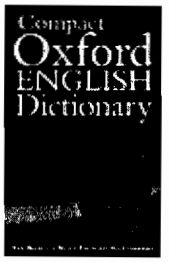
Clerks and Typists

Maude L. Barnes • Esther L. Gauthier • Mildred A. Lindsay • Maureen E. McCartney • Mildred M. McWha • Catherine T. Meaney • Frances W. Muldrew • Mildred C. Paquette • Genevieve M. Sherry • Francine A. Socha

[3]bottom *adj* 1 : of, relating to, or situated at the bottom ⟨~ rock⟩ 2 : frequenting the bottom ⟨~ fishes⟩
bot·tom-land \'bät-əm-ˌland\ *n* : BOTTOM 5
bot·tom-less \-ləs\ *adj* 1 : having no bottom ⟨a ~ chair⟩ 2 a : extremely deep b : impossible to comprehend : UNFATHOMABLE ⟨a ~ mystery⟩ c : BOUNDLESS, UNLIMITED 3 a [fr. the absence of lower as well as upper garments] : NUDE ⟨~ dancers⟩ b : featuring nude entertainers ⟨a ~ nightclub⟩ — **bot·tom-less-ly** *adv* — **bot·tom-less-ness** *n*
bot·tom·most \'bät-əm-ˌmōst\ *adj* 1 a : situated at the very bottom : LOWEST, DEEPEST b : LAST ⟨the ~ part of the day —Alfred Kazin⟩ 2 : most basic ⟨the ~ problems facing the world⟩
bottom out *vi, of a security market* : to decline to a point where demand begins to exceed supply and a rise in prices is imminent
bottom round *n* : meat (as steak) from the outer part of a round of beef
bot·tom·ry \'bät-əm-rē\ *n, pl* **-ries** [modif. of D *bodemerij*, fr. *bodem* bottom, ship; akin to OHG *bodam*] : a contract by which a ship is hypothecated as security for repayment of a loan at the end of a successful voyage
bot·u·lin \'bäch-ə-lən\ *n* [prob. fr. NL *botulinus*] : a toxin that is formed by the botulinum and is the direct cause of botulism
bot·u·li·num \ˌbäch-ə-'lī-nəm\ *also* **bot·u·li·nus** \-nəs\ *n* [NL, fr. L *botulus* sausage] : a spore-forming bacterium (*Clostridium botulinum*) that secretes botulin — **bot·u·li·nal** \-'līn-ᵊl\ *adj*
bot·u·lism \'bäch-ə-ˌliz-əm\ *n* : acute food poisoning caused by botulin in food
bou·clé *or* **bou·cle** \bü-'klā\ *n* [F *bouclé* curly, fr. pp. of *boucler* to curl, fr. *bocle* buckle, curl] 1 : an uneven yarn of three plies one of which forms loops at intervals 2 : a textile fabric of bouclé yarn
bou·doir \'büd-ˌwär, 'bůd-\ *n* [F, fr. *bouder* to pout] : a woman's dressing room, bedroom, or private sitting room
bouf·fant \bü-'fänt, 'bü-ˌ\ *adj* [F, fr. MF, fr. prp. of *bouffer* to puff] : puffed out ⟨~ hairdos⟩ ⟨a ~ veil⟩
bou·gain·vil·lea *or* **bou·gain·vil·laea** \ˌbüg-ən-'vil-yə, ˌbōg-, ˌbůg-, -'vē-(y)ə\ *n* [NL, fr. Louis Antoine de *Bougainville*] : any of a genus (*Bougainvillaea*) of the four-o'clock family of ornamental tropical American woody vines with brilliant purple or red floral bracts
bough \'baů\ *n* [ME, shoulder, bough, fr. OE *bōg*; akin to OHG *buog* shoulder, Gk *pēchys* forearm] : a branch of a tree; *esp* : a main branch — **boughed** \'baůd\ *adj*
bought \'bȯt\ *adj* [pp. of *buy*] : READY-MADE ⟨~ clothes⟩
bought-en \-ᵊn\ *adj* [*bought* + *-en* (as in *forgotten*)] *chiefly dial* : BOUGHT ⟨the only ~ carpet in the region —H. W. Thompson⟩
bou·gie \'bü-ˌzhē, -ˌjē\ *n* [F, fr. *Bougie*, seaport in Algeria] 1 : a wax candle 2 a : a tapering cylindrical instrument for introduction into a tubular passage of the body b : SUPPOSITORY
bouil·la·baisse \ˌbů-yə-'bās\ *n* [F] : a highly seasoned fish stew made of at least two kinds of fish
bouil·lon \'bůl-ˌyän, -yən; 'bü-yän, 'bů-\ *n* [F, fr. OF *boillon*, fr. *boillir* to boil] : a clear seasoned soup made usu. from lean beef
bouillon cube *n* : a cube of evaporated seasoned meat extract
boul·der \'bōl-dər\ *n* [short for *boulder stone*, fr. ME *bulder ston*, part trans. of a word of Scand origin; akin to Sw dial. *bullersten* large stone in a stream, fr. *buller* noise + *sten* stone] : a detached and rounded or much-worn mass of rock — **boul·dered** \-dərd\ *adj* — **boul·dery** \-d(ə-)rē\ *adj*
[1]bou·le \'bü-(ˌ)lē, bü-'lā\ *n* [Gk *boulē*, lit., will, fr. *boulesthai* to wish] : a legislative council of ancient Greece consisting first of an aristocratic advisory body and later of a representative senate
[2]boule \'bül\ *n* [F, ball — more at BOWL] : a pear-shaped mass (as of sapphire) formed synthetically in a special furnace with the atomic structure of a single crystal
bou·le·vard \'bůl-ə-ˌvärd, 'bül-\ *n* [F, modif. of MD *bolwerc* bulwark] : a broad often landscaped thoroughfare
bou·le·vard·ier \ˌbůl-ə-ˌvär-'dyā, ˌbül-, -'di(ə)r\ *n* [F, fr. *boulevard* + *-ier* -er] : a frequenter of the Parisian boulevards; *broadly* : MAN-ABOUT-TOWN
bou·le·ver·se·ment \bül-(ə-)ver-sə-mäⁿ\ *n* [F] 1 : REVERSAL 2 : a violent disturbance : DISORDER
boulle \'bül, 'byü(ə)l\ *n* [André Charles *Boulle* †1732 F cabinetmaker] : inlaid decoration of tortoiseshell, yellow metal, and white metal in cabinetwork
[1]bounce \'baůn(t)s\ *vb* **bounced; bounc·ing** [ME *bounsen*] *vt* 1 *obs* : BEAT, BUMP 2 : to cause to rebound ⟨~ a ball⟩ 3 a : DISMISS, FIRE b : to expel precipitately from a place ~ *vi* 1 : to rebound after striking 2 : to recover from a blow or a defeat quickly — usu. used with *back* 3 : to be returned by a bank as no good ⟨his checks ~⟩ 4 a : to leap suddenly : BOUND b : to walk with springing steps 5 : to hit a baseball so that it hits the ground before it reaches an infielder
[2]bounce *n* 1 a : a sudden leap or bound b : REBOUND 2 : BLUSTER 3 : VERVE, LIVELINESS
bounc·er \'baůn(t)-sər\ *n* : one that bounces: as a : one employed to restrain or eject disorderly persons b : a batted baseball that bounces
bounc·ing \-siŋ\ *adj* 1 : enjoying good health : ROBUST 2 : LIVELY, ANIMATED — **bounc·ing-ly** \-siŋ-lē\ *adv*
bouncing bet \-'bet\ *n, often cap 2d B* [fr. *Bet*, nickname for *Elizabeth*] : a European perennial herb (*Saponaria officinalis*) of the pink family that is widely naturalized in the U.S. and has pink or white flowers and leaves which yield a detergent when bruised — called also *soapwort*
bouncy \'baůn(t)-sē\ *adj* **bounc·i·er; -est** 1 : BUOYANT, EXUBERANT 2 : RESILIENT 3 : marked by or producing bounces — **bounc·i·ly** \-sə-lē\ *adv*
[1]bound \'baůnd\ *adj* [ME *boun*, fr. ON *būinn*, pp. of *būa* to dwell, prepare; akin to OHG *būan* to dwell — more at BOWER] 1 *archaic* : READY 2 : intending to go : GOING ⟨~ for home⟩ ⟨college-*bound*⟩
[2]bound *n* [ME, fr. OF *bodne*, fr. ML *bodina*] 1 a : a limiting line : BOUNDARY — usu. used in pl. b : something that limits or restrains ⟨beyond the ~s of decency⟩ 2 *usu pl* a : BORDERLAND b : the land within certain bounds
[3]bound *vt* 1 : to set limits to : CONFINE 2 : to form the boundary of : ENCLOSE 3 : to name the boundaries of
[4]bound *adj* [ME *bounden*, fr. pp. of *binden* to bind] 1 a : fastened by or as if by a band : CONFINED ⟨desk-*bound*⟩ b : CERTAIN, SURE ⟨~ to rain soon⟩ 2 : placed under legal or moral restraint or obligation : OBLIGED ⟨duty-*bound*⟩ 3 : made costive : CONSTIPATED 4 *of a book* a : secured to the covers by cords or tapes b : cased in 5 : DETERMINED, RESOLVED 6 : held in chemical or physical combination ⟨~ water in a molecule⟩ 7 : always occurring in combination with another linguistic form ⟨*un-* in *unknown* and *-er* in *speaker* are ~ forms⟩ — compare FREE
[5]bound *n* [MF *bond*, fr. *bondir* to leap, fr. (assumed) VL *bombitire* to hum, fr. L *bombus* deep hollow sound — more at BOMB] 1 : LEAP, JUMP 2 : the action of rebounding : BOUNCE
[6]bound *vi* 1 : to move by leaping 2 : REBOUND, BOUNCE
bound·ary \'baůn-d(ə-)rē\ *n, pl* **-aries** : something that indicates or fixes a limit or extent; *specif* : a bounding or separating line
boundary layer *n* : a region of retarded fluid near the surface of a body which moves through a fluid or past which a fluid moves
bound·en \'baůn-dən\ *adj* [ME] 1 *archaic* : being under obligation : BEHOLDEN 2 : made obligatory : BINDING ⟨our ~ duty⟩
bound·er \-dər\ *n* 1 : one that bounds 2 : a man of objectionable social behavior : CAD
bound·er-ish \-də-rish\ *adj* : resembling or typical of a bounder — **bound·er-ish-ly** *adv*
bound·less \'baůn-dləs\ *adj* : having no boundaries : VAST — **bound·less-ly** *adv* — **bound·less-ness** *n*
bound up *adj* : closely involved or associated — usu. used with *with*
boun·te·ous \'baůnt-ē-əs\ *adj* [ME *bountevous*, fr. MF *bontif* kind, fr. OF, fr. *bonté*] 1 : giving or disposed to give freely 2 : liberally bestowed — **boun·te·ous-ly** *adv* — **boun·te·ous-ness** *n*
boun·tied \'baůnt-ēd\ *adj* 1 : having the benefit of a bounty 2 : rewarded or rewardable by a bounty
boun·ti·ful \'baůnt-i-fəl\ *adj* 1 : liberal in bestowing gifts or favors 2 : given or provided abundantly : PLENTIFUL ⟨a ~ harvest⟩ *syn* see LIBERAL *ant* niggardly — **boun·ti·ful-ly** \-f(ə-)lē\ *adv* — **boun·ti·ful-ness** \-fəl-nəs\ *n*
boun·ty \'baůnt-ē\ *n, pl* **bounties** [ME *bounte* goodness, fr. OF *bonté*, fr. L *bonitat-, bonitas*, fr. *bonus* good, fr. OL *duenos*; akin to MHG *zwiden* to grant, L *bene* well] 1 : liberality in giving : GENEROSITY 2 : something that is given generously 3 : yield esp. of a crop 4 : a reward, premium, or subsidy esp. when offered or given by a government: as a : an extra allowance to induce entry into the armed services b : a grant to encourage an industry c : a payment to encourage the destruction of noxious animals d : a payment for the capture of an outlaw
bounty hunter *n* 1 : one that hunts predatory animals for the reward offered 2 : one that tracks down and captures outlaws for whom a reward is offered
bou·quet \bō-'kā, bü-\ *n* [F, fr. MF, thicket, fr. ONF *bosquet*, fr. OF *bosc* forest — more at BOSCAGE] 1 a : flowers picked and fastened together in a bunch : NOSEGAY b : a large flight of fireworks 2 : COMPLIMENT 3 a : a distinctive and characteristic fragrance (as of wine) b : a subtle aroma or quality (as of an artistic performance or a piece of writing) *syn* see FRAGRANCE
bour·bon \'bů(ə)r-bən, 'bō(ə)r-, 'bō(ə)r-; *usu* 'bər- *in sense 4*\ *n* [*Bourbon*, seigniory in France] 1 *cap* : a member of a French family founded in 1272 to which belong the rulers of France from 1589 to 1793 and from 1814 to 1830, of Spain from 1700 to 1808, from 1814 to 1868, from 1875 to 1931, and from 1975, of Naples from 1735 to 1805, and of the Two Sicilies from 1815 to 1860 2 *often cap* : a person who clings obstinately to the social and political ideas of the old order of things; *specif* : an extremely conservative member of the U.S. Democratic party usu. from the South 3 [*Bourbon* (now Réunion), French island in the Indian ocean] : a rose (*Rosa borboniana*) of compact upright growth with shining leaves, prickly branches, and clustered flowers 4 [*Bourbon* county, Kentucky] : a whiskey distilled from a mash made up of not less than 51 percent corn plus malt and rye — compare CORN WHISKEY — **bour·bon-ism** \-bə-ˌniz-əm\ *n, often cap*
bour·don \'bů(ə)rd-ᵊn\ *n* [ME *burdoun*, fr. MF *bourdon* bass pipe, of imit. origin] : a drone bass (as in a bagpipe)
bourg \'bů(ə)r(g)\ *n* [ME, fr. MF, fr. OF *borc*, fr. L *burgus* fortified place, of Gmc origin; akin to OHG *burg* fortified place — more at BOROUGH] : TOWN, VILLAGE: as a : one neighboring a castle b : a market town
[1]bour·geois \'bů(ə)rzh-ˌwä, bůrzh-\ *n, pl* **bourgeois** \-ˌwä(z), -'wä(z)\ [MF, fr. OF *borjois*, fr. *borc*] 1 a : BURGHER b : a middle-class person 2 : one with social behavior and political views held to be influenced by private-property interest : CAPITALIST 3 *pl* : BOURGEOISIE
[2]bourgeois *adj* 1 : of, relating to, or characteristic of the townsman or of the social middle class 2 : marked by a concern for material interests and respectability and a tendency toward mediocrity 3 : dominated by commercial and industrial interests : CAPITALISTIC — **bour·geois-ify** \bůrzh-'wäz-ə-ˌfī\ *vb*
bour·geoise \'bů(ə)rzh-ˌwäz, bůrzh-\ *n* [F, fem. of *bourgeois*] 1 : a woman of the middle class 2 : BOURGEOIS
bour·geoi·sie \ˌbůrzh-ˌwä-'zē\ *n* [F, fr. *bourgeois*] 1 : MIDDLE CLASS 2 : a social order dominated by bourgeois
bour·geon \'bər-jən\ *var of* BURGEON
[1]bourn *or* **bourne** \'bō(ə)rn, 'bó(ə)rn, 'bů(ə)rn\ *n* [ME *burn*, *bourne* — more at BURN] : STREAM, BROOK
[2]bourn *or* **bourne** *n* [MF *bourne*, fr. OF *bodne* — more at BOUND] 1 *archaic* : BOUNDARY, LIMIT 2 *archaic* : GOAL, DESTINATION

TAB 30

Compact Oxford English Dictionary



limit

• **noun 1** a point beyond which something does not or may not pass. **2** a restriction on the size or amount of something. **3** the furthest extent of one's endurance.

• **verb** (**limited**, **limiting**) set or serve as a limit to.

— PHRASES **be the limit** informal be intolerable. **off limits** out of bounds. **within limits** up to a point.

— DERIVATIVES **limiter** noun **limitless** adjective.

— ORIGIN Latin *limes* 'boundary, frontier'.

Perform another search of the Compact Oxford English Dictionary

- Ask The Experts
- Better Writing
- World of Words
- Games
- Global English
- Foreign Languages

LINKS

→ AskOxford Shop
→ External Web Links
→ OUP Website
→ Children's Dictionaries
→ ELT Dictionaries
→ Oxford Online

About this dictionary
The *Compact Oxford English Dictionary of Current English* contains 145,000 words, phrases, and definitions.
Find out more about Oxford's range of English dictionaries
Sign up for the AskOxford Word of the Day

Search the Little Oxford Dictionary of Quotations
Search the Concise Dictionary of First Names

TAB 31

The American Heritage® Dictionary of the English Language: Fourth Edition. 2000.

limit

SYLLABICATION: lim·it

PRONUNCIATION: lĭm'ĭt

NOUN: **1.** The point, edge, or line beyond which something cannot or may not proceed. **2. limits** The boundary surrounding a specific area; bounds: *within the city limits.* **3.** A confining or restricting object, agent, or influence. **4.** The greatest or least amount, number, or extent allowed or possible: *a withdrawal limit of $200; no minimum age limit.* **5.** *Games* The largest amount which may be bet at one time in games of chance. **6.** *abbr.* **lim** *Mathematics* A number or point *L* that is approached by a function *f(x)* as *x* approaches *a* if, for every positive number ε, there exists a number δ such that $|f(x)-L| < \varepsilon$ if $0 < |x-a| < \delta$. Also called *limit point*, *point of accumulation*. **7.** *Informal* One that approaches or exceeds certain limits, as of credibility, forbearance, or acceptability: *He is the limit of irresponsibility.*

TRANSITIVE VERB: Inflected forms: **lim·it·ed, lim·it·ing, lim·its**
1. To confine or restrict within a boundary or bounds. **2.** To fix definitely; to specify.

ETYMOLOGY: Middle English *limite*, from Old French, border, from Latin *līmes*, *līmit-*, border, limit.

OTHER FORMS: **lim'it·a·ble** —ADJECTIVE

SYNONYMS: *limit*, *restrict*, *confine*, *circumscribe* These verbs mean to establish or keep within specified bounds. *Limit* refers principally to the establishment of a maximum beyond which a person or thing cannot or may not go: *The Constitution limits the President's term of office to four years.* To *restrict* is to keep within prescribed limits, as of choice or action: *The sale of alcoholic beverages is restricted to those over 21. Confine* suggests imprisonment, restraint, or impediment: *The children were confined to the nursery. Circumscribe* connotes an encircling or surrounding line that confines, especially narrowly: *"A man . . . should not circumscribe his activity by any inflexible fence of rigid rules"* (John Stuart Blackie).

CONTENTS · INDEX · ILLUSTRATIONS · BIBLIOGRAPHIC RECORD

‹ limina limitary ›

Google Search

Click here to shop the Bartleby Bookstore.

Welcome · Press · Advertising · Linking · Terms of Use · © 2005 Bartleby.com

TAB 32

US006099580A

United States Patent [19]

Boyle et al.

[11] **Patent Number: 6,099,580**

[45] **Date of Patent: Aug. 8, 2000**

[54] **METHOD FOR PROVIDING PERFORMANCE-DRIVEN LOGIC OPTIMIZATION IN AN INTEGRATED CIRCUIT LAYOUT DESIGN**

[75] Inventors: **Douglas B. Boyle**, Palo Alto; **James S. Koford**, San Jose, both of Calif.

[73] Assignee: **Monterey Design Systems, Inc.**, Sunnyvale, Calif.

[21] Appl. No.: **09/021,973**

[22] Filed: **Feb. 11, 1998**

[51] **Int. Cl.**[7] **G06F 17/50**

[52] **U.S. Cl.** **716/7**; 716/8

[58] **Field of Search** 395/500.08, 500.09, 395/500.1, 500.11, 500.12; 716/7, 8

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,495,419	2/1996	Rostoker et al.	700/121
5,526,517	6/1996	Jones et al.	707/8
5,875,117	2/1999	Jones et al.	716/14
5,909,376	6/1999	Scepanovic et al.	716/8
5,914,887	6/1999	Scepanovic et al.	716/8

FOREIGN PATENT DOCUMENTS

0 463 301 A2 1/1992 European Pat. Off. .

OTHER PUBLICATIONS

Sato et al "Post–Layout Optimization for Deep Submicron Design," ACM, Jan. 1996, pp. 1–6.

Kannan et al "A Methodoligy and Algorithms for Post–Placement Delay Optimization," ACM/IEEE, 1994, pp. 327–332.

Chandrasekhar et al "Effective Coupling Between Logic Synthesis and Layout Tools for Synthesis..Circuits," VLSI Design, 1997, vol. 5, No. 2, pp. 125–140.

Wiederhold et al "Deep–Submicron ASIC Design Requires Design Planning," EDN, Feb. 1995, pp. 95–100.

Primary Examiner—Paul R. Lintz

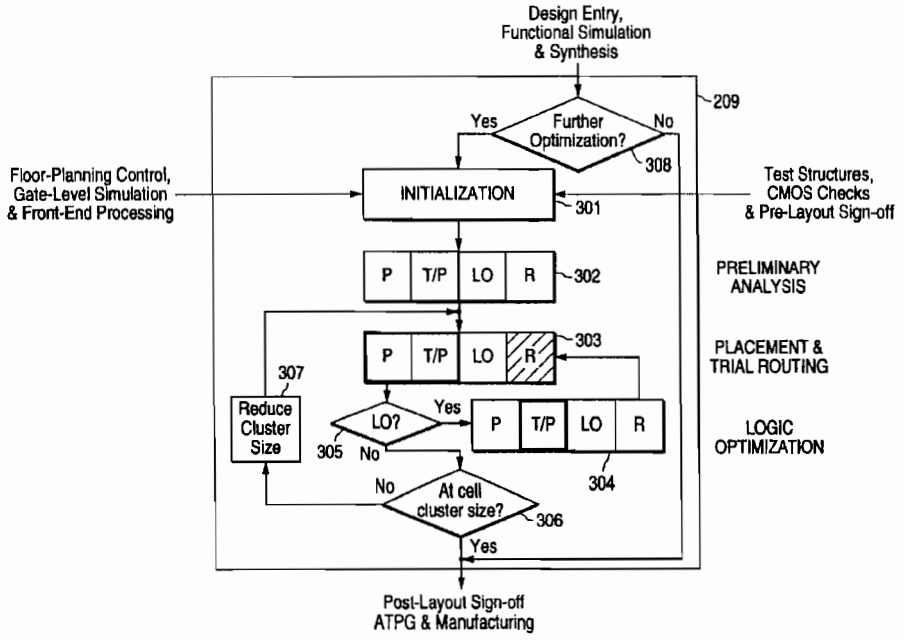
Assistant Examiner—Vuthe Siek

Attorney, Agent, or Firm—Skjerven, Morrill, MacPherson, Franklin & Friel LLP; Edward C. Kwok

[57] **ABSTRACT**

A method for optimizing layout design using logical and physical information performs placement, logic optimization and routing and routing estimates concurrently. In one embodiment, circuit elements of the integrated circuit is partitioned into clusters. The clusters are then placed and routed by iterating over an inner-loop and an outer-loop according to cost functions in the placement model which takes into consideration interconnect wiring delays. Iterating over the inner-loop, logic optimization steps improves the cost functions of the layout design. Iterating over the outer-loop, the size of the clusters, hence the granularity of the placement, is refined until the level of individual cells is reached. The present method is especially suited for parallel processing by multiple central processing units accessing a shared memory containing the design data base.

39 Claims, 6 Drawing Sheets



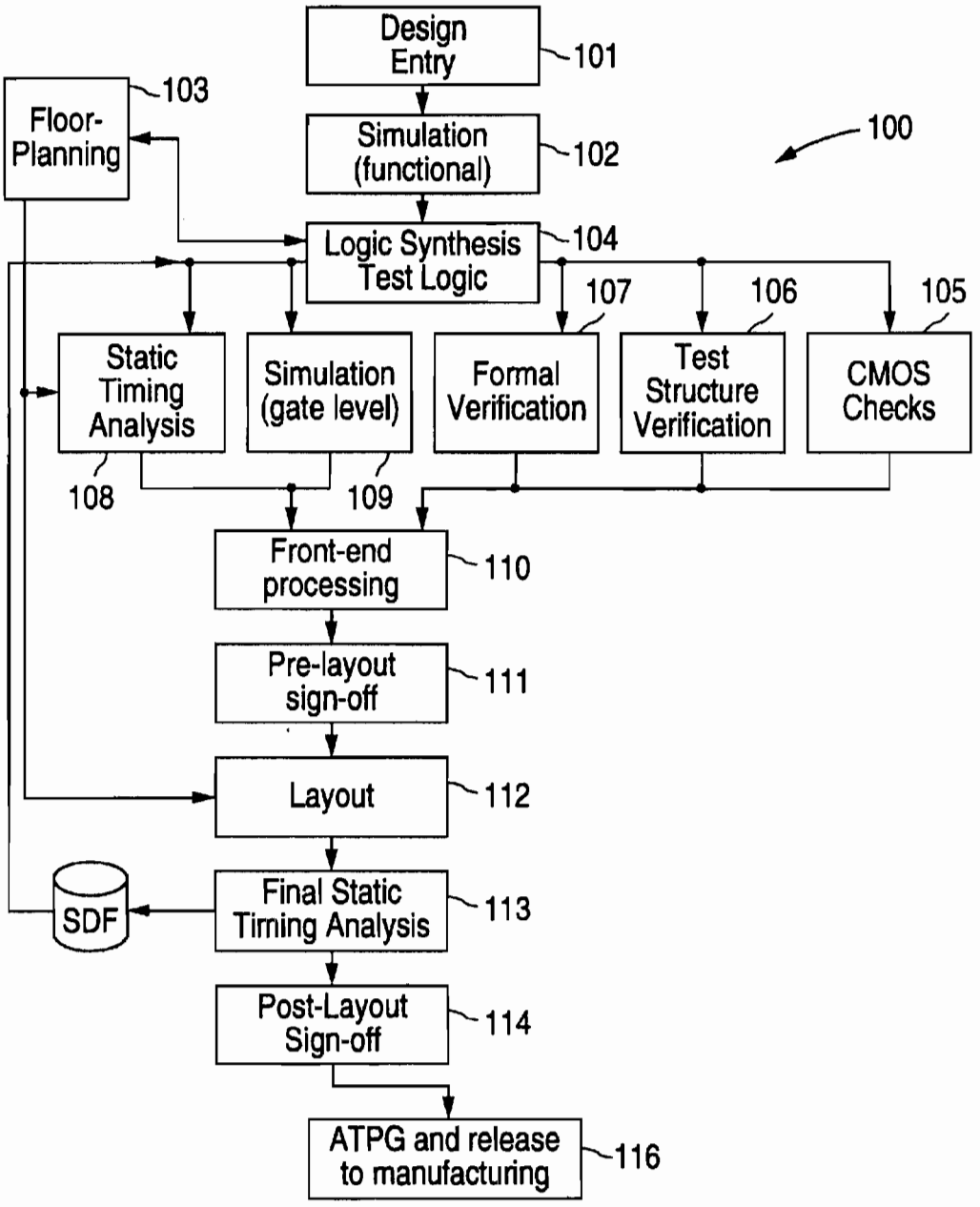


FIGURE 1
(PRIOR ART)

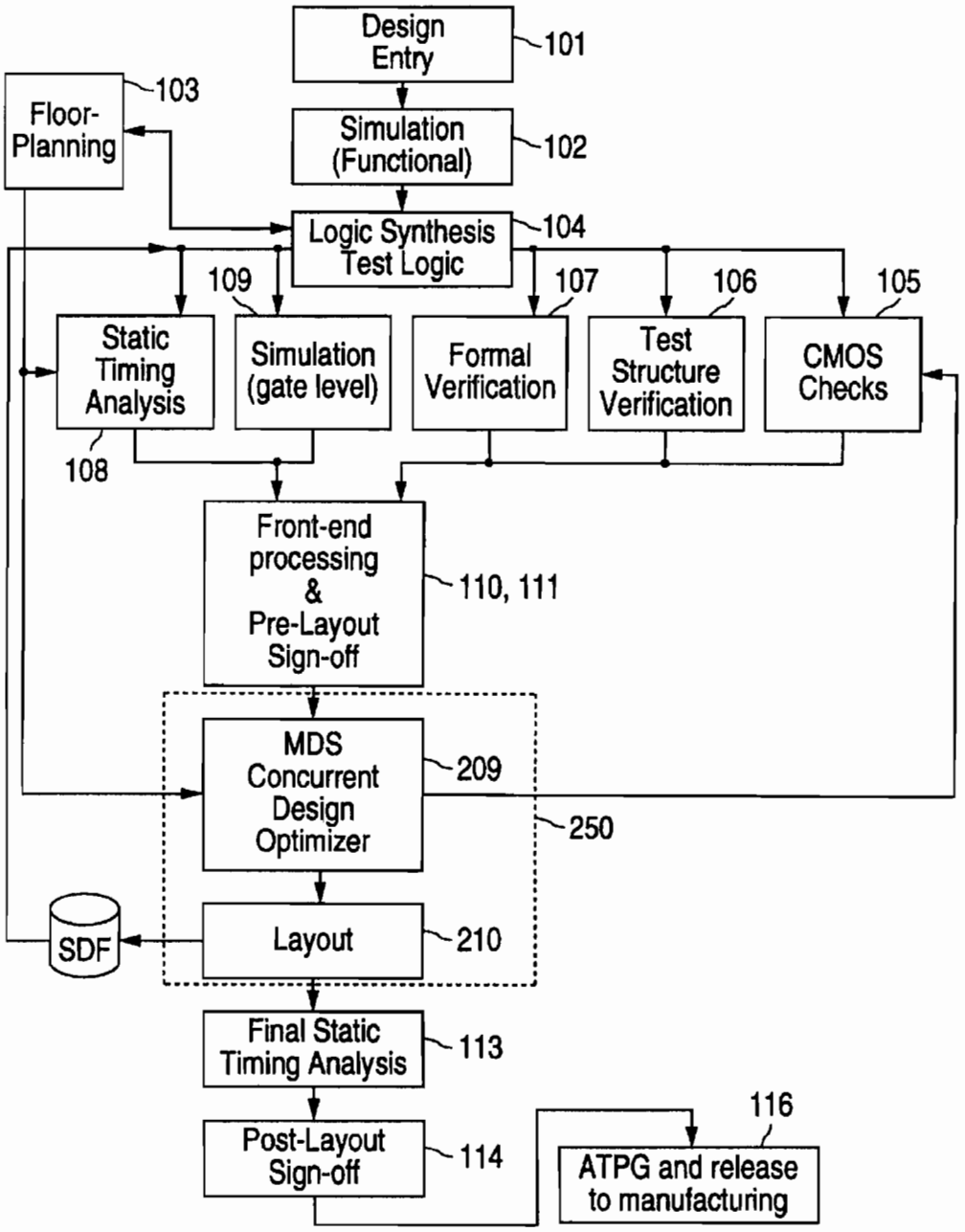

Design Entry
101
Floor-Planning
103
Simulation (Functional)
102
Logic Synthesis Test Logic
104
109
107
106
105
Static Timing Analysis
Simulation (gate level)
Formal Verification
Test Structure Verification
CMOS Checks
108
Front-end processing & Pre-Layout Sign-off
110, 111
MDS Concurrent Design Optimizer
209
250
SDF
Layout
210
Final Static Timing Analysis
113
Post-Layout Sign-off
114
116
ATPG and release to manufacturing

FIGURE 2

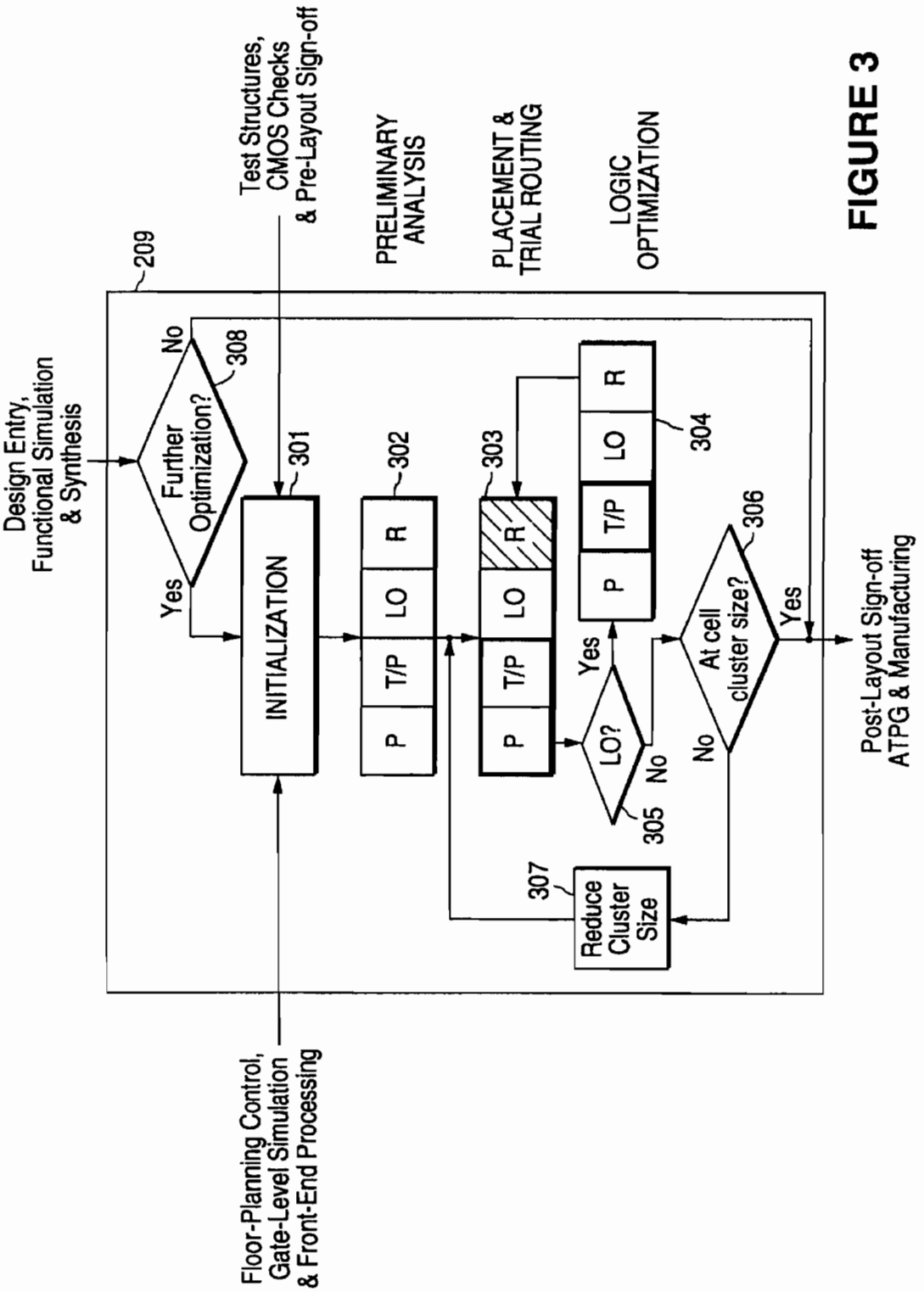

Design Entry, Functional Simulation & Synthesis
209
Further Optimization?
No
Yes
308
INITIALIZATION
301
Test Structures, CMOS Checks & Pre-Layout Sign-off
Floor-Planning Control, Gate-Level Simulation & Front-End Processing
PRELIMINARY ANALYSIS
302
P
T/P
LO
R
PLACEMENT & TRIAL ROUTING
303
LOGIC OPTIMIZATION
304
LO?
Yes
No
305
At cell cluster size?
306
Yes
No
307
Reduce Cluster Size
Post-Layout Sign-off ATPG & Manufacturing

FIGURE 3

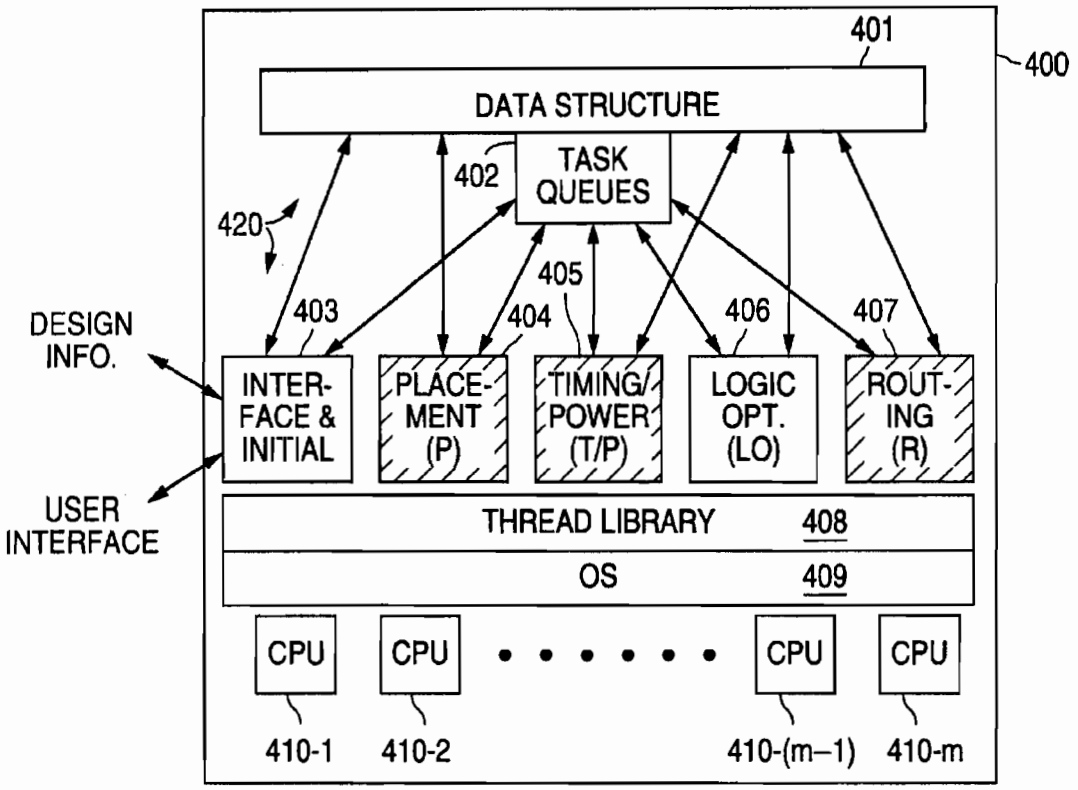


FIGURE 4

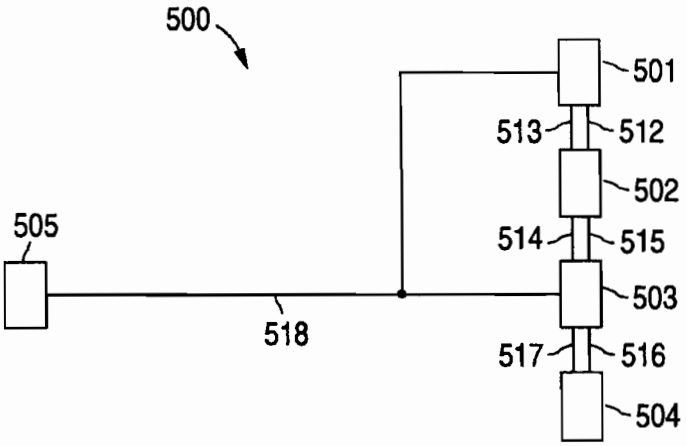


FIGURE 5(a)

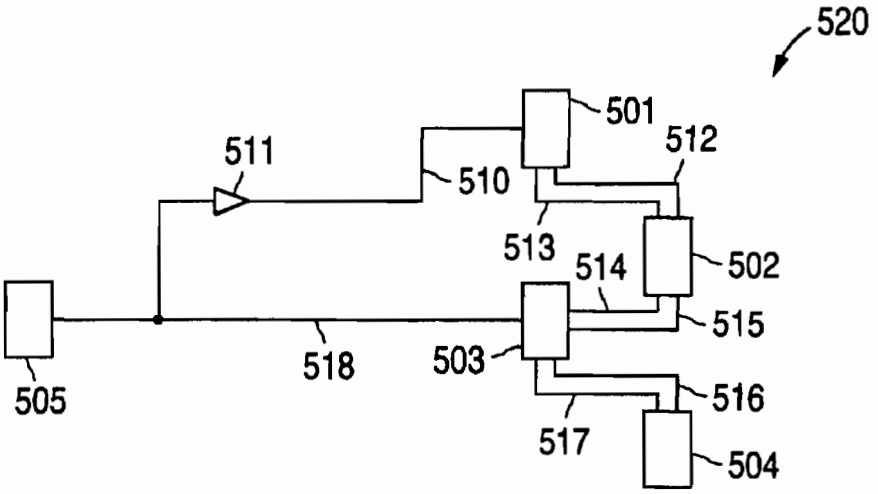


FIGURE 5(b)
(PRIOR ART)

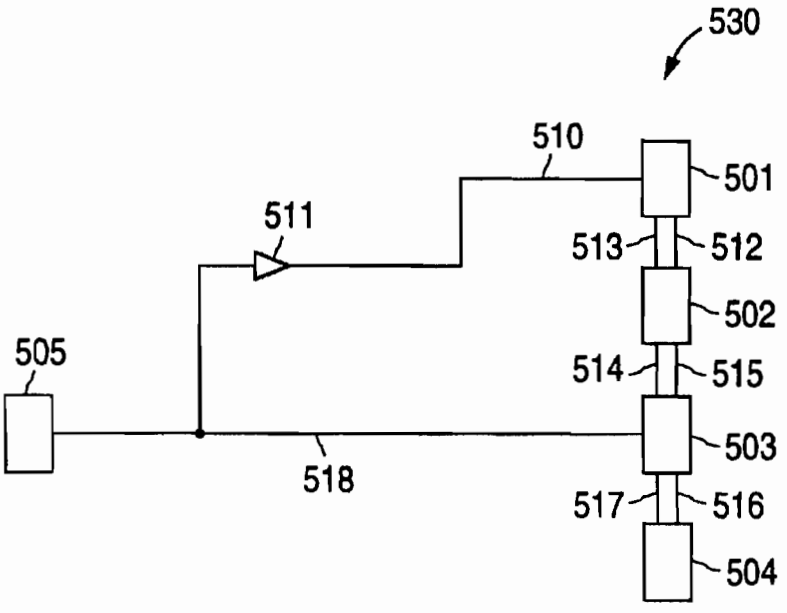


FIGURE 5(c)

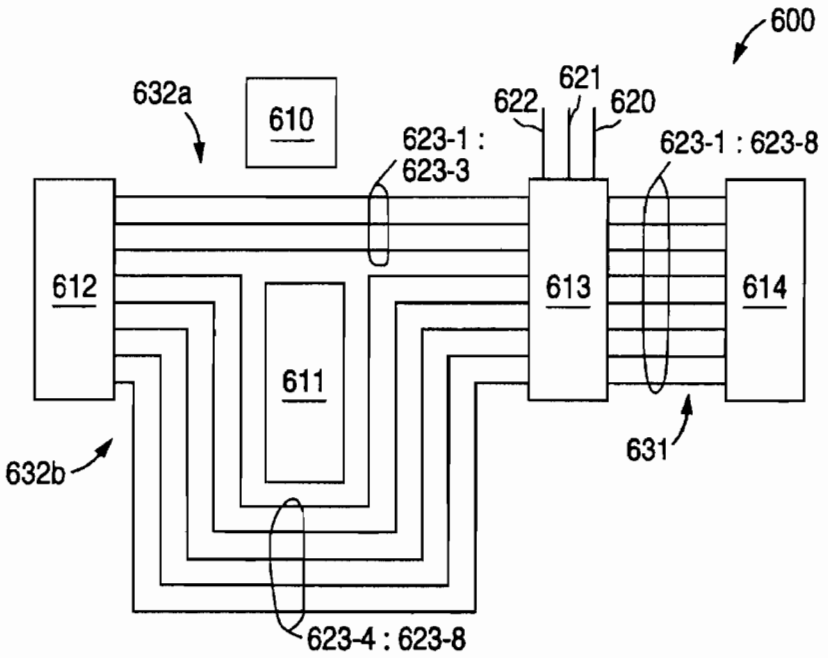


FIGURE 6(a)
(PRIOR ART)

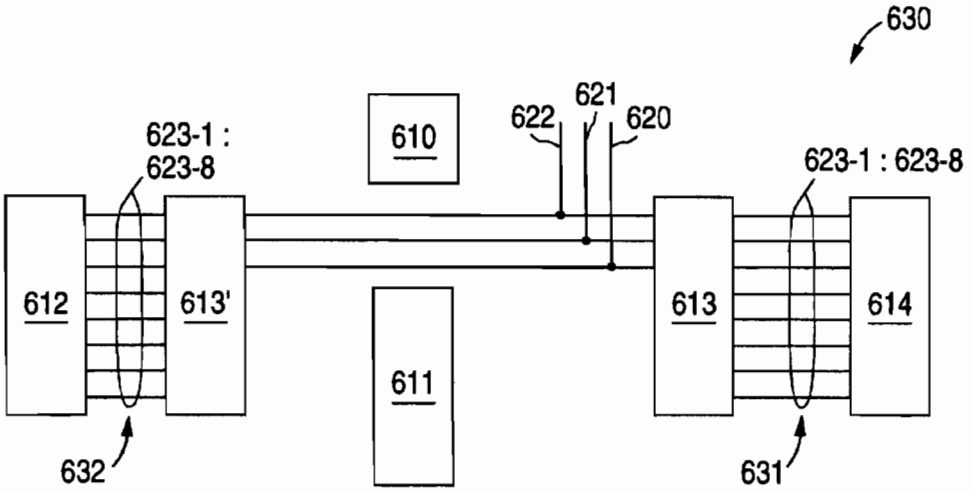


FIGURE 6(b)

METHOD FOR PROVIDING PERFORMANCE-DRIVEN LOGIC OPTIMIZATION IN AN INTEGRATED CIRCUIT LAYOUT DESIGN

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to integrated circuit physical design automation. In particular, the present invention relates to an automated method and system for providing a physical design of an integrated circuit that is optimized for meeting power and interconnect-dominant timing requirements.

2. Discussion of the Related Art

Existing top-down design methods focus on optimizing transistors and gates, and model interconnect lines as merely "parasitic" elements of these transistors and gates. Implicit in this view of interconnect lines is the assumption that wiring delays and congestion in the interconnect lines are secondary to transistors and gates, and therefore can be taken into account as corrections to the timing and density models of the transistor and gates. As feature sizes of integrated circuits continue to shrink, this assumption is clearly no longer valid. In fact, interconnect is expected to dominate both performance and density in the near future.

FIG. **1** shows a typical existing design method **100**. As shown in FIG. **1**, an integrated circuit design is captured by a design entry step **101**. Design entry step **101** is typically facilitated by a design capture system allowing the user to specify the logic design of the integrated circuit graphically, through a hardware description language (e.g., VHDL), or both. Typically, at design entry step **101**, the user need not specify all elements of the design at the logic gate level. Many elements can be specified at a higher functional level (e.g., register-transfer level).

Upon completion of design entry step **101**, a functional simulation step **102** is typically carried out using a functional simulator to verify functional behavior of the design. Based on a verified functional level description ("functional design"), the functional design can then be synthesized to the logic gate level in logic synthesis step **104**, using a logic synthesis tool. Typically, at logic synthesis step **104**, additional circuits for diagnostic and test purposes are included to provide a synthesized logic circuit ("gate-level design"). Such additional circuits may include, for example, test circuits for a boundary scan design under the JTAG standard.

Using the gate-level design, an initial circuit partitioning can be performed, i.e., floor planning step **103**, to allow a first estimate of circuit size and to group highly connected portions of the design together to facilitate a subsequent layout step. In addition, a logic simulation (simulation step **109**) and a static timing analysis (timing analysis step **108**) are performed on the gate-level design to verify the gate-level design's functional behavior, and to extract estimates of timing parameters.

During this time, at formal verification step **107** and test structure verification step **106**, the gate-level design is verified against both the functional design and a behavior description of the additional included circuits to ensure both the functional design's and the test structure's behaviors are preserved. Further, the gate-level design is also checked, at technology checking step **105**, that technology-specific logic design rules are not violated.

The gate-level design, together with timing information from the static timing analysis, are integrated into a pre-layout design database in front-end processing step **110**. At this time, a pre-layout signoff step **111** signifies the beginning of the physical realization phase of the integrated circuit design. FIG. **1** shows a layout step **112** in which the physical realization ("layout") is created by performing a number of tasks ("layout design tasks") iteratively.

Typically, layout design tasks include, generally, the steps of circuit partitioning, placement and routing. As mentioned above, an initial circuit partition based on the gate-level design is already provided at floor-planning step **103**. Based on this initial partition, the circuit partitioning step in layout step **112** further refines circuit partitions down to the level of individual "cells" (e.g., logic gates or macro cells). These cells are then placed according to some constraints, which are typically expressed by a cost function. Typical constraints relate to area, power and local timing. The cells so placed are then routed to provide the necessary interconnect. The routing is also typically performed according to certain constraints, such as local timing and power constraints.

In the prior art, a final static timing analysis step **113** is then performed on the routed layout design, incorporating into the timing information delays introduced by the routing step. If final static timing analysis step **113** uncovers timing problems in some signal paths, an optimization cycle is initiated by flagging the gates along the problematic signal paths and returning the gate-level design back to logic synthesis step **104**. In logic synthesis step **104**, logic synthesis techniques are applied to improve the gate-level design in a revised gate-level design. Steps **105–113** are then repeated on the revised gate-level design. This optimization cycle is repeated until all timing problems are resolved, represented by the post-layout sign-off step **114**. Test patterns can then be generated in an automatic test pattern generation (ATPG) step **116**, and the final layout design can then be manufactured.

The existing top-down design methods not only suffer from the defective interconnect model mentioned above, but also from the long elapsed time between optimization cycles. With each optimization cycle, the logic synthesis, circuit partitioning, placement and routing are "point" tools, each operating on the entire design. Such loose coupling between tools in this optimization cycle is inefficient, since the steps towards convergence to an optimized layout tend to be small for each optimization cycle. For example, any possible improvement realizable by resynthesis of the logic circuit that arises because of a different placement of cells cannot be taken advantage of until the next optimization cycle. Furthermore, at the current and future circuit densities, the amount of CPU cycles used in each point tool in the optimization cycle is enormous, with many cycles expended in repeating tasks also performed in the previous optimization cycles. Further, since each existing point tool is typically acquired from a different vendor, there is substantial inefficiency in the interface between point tools. Typically, each point tool reads into memory a large data file provided by an upstream point tool, and provides its results in another large data file to be read by a downstream point tool.

The long optimization cycles can be alleviated somewhat by the use of parallel point tools. An example of such a parallel point tool is disclosed in U.S. Pat. No. 5,495,419 to Rostoker et al., entitled "INTEGRATED CIRCUIT PHYSICAL DESIGN AUTOMATIC SYSTEM UTILIZING OPTIMIZATION PROCESS DECOMPOSITION AND PARALLEL PROCESSING," issued Feb. 27, 1996. In Rostoker et al, a parallel placement algorithm is executed by parallel processors, thereby shortening the time to achieve the placement function.

Thus, a method is desired for optimizing the integrated circuit design process, which both takes advantage of parallel algorithms and closely couples the layout design tasks in the optimization cycle is desired.

SUMMARY OF THE INVENTION

The present invention provides a method for optimizing a layout design, which minimizes the optimization cycle by incorporating interconnect wiring delays and performing logic optimization in the placement and routing operations. In one embodiment of the present invention, the method includes the steps of: (a) partitioning circuit elements of the layout design into clusters; (b) mapping a portion of the layout design to each cluster, so as to obtain a first placement; (c) providing routing between circuit elements within each cluster and between circuit elements of different clusters; (d) performing a timing analysis to provide estimates of interconnect delay between circuit elements within each cluster and between circuit elements of different clusters; (e) performing a logic optimization operation on a selected one of the clusters to obtain a second gate-level design based on a cost function. The steps (a)–(e) are reiterated until the cost function becomes less than a predetermined threshold. The clustering of circuit elements allow circuit elements having high connectivity amongst them to be kept together and thus placed in close proximity of each other, while minimizing inter-cluster interconnect wiring delays.

In that embodiment, the method is iterated over an outer-loop in which each iteration divides each cluster into smaller subclusters and applies steps (b) through (f) on the subclusters until the subcluster includes only primitive circuit elements.

In one embodiment, the logic optimization operation, which is performed concurrently with the placement and routing operations, (i) reassigns circuit elements between clusters, (ii) inserts or deletes signal buffers between circuit elements (iii) synthesizes alternative implementations of logic functions.

The method of the present invention can be implemented in a parallel processing design automation system to achieve high performance. Such a parallel processing system includes (a) multiple central processing units (CPUs) executing independently of each other; (b) a shared memory accessible by each CPU for holding data structures of the integrated circuit design; and (c) a control program managing a task list which specifies tasks for optimizing the integrated design. The control program assigns tasks on the task list to each of the CPUs. The tasks include placement, performance analysis, logic optimization and routing. The control program provides a locking mechanism for locking data structures stored in the shared memory to support concurrent operations by the CPUs each accessing the integrated circuit design represented by the data structures. Further, the parallel processing design automation system supports multithread execution. Parallelism is further enhanced by adopting algorithms for placement, timing analysis, logic optimization and routing suitable for parallel execution by the multiple CPUs each accessing the design data structures stored in the shared memory.

The present invention is better understood upon consideration of the detailed description below in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** shows an existing method **100** for designing an integrated circuit in the prior art.

FIG. **2** shows a method **200** for designing an integrated circuit in one embodiment of the present invention.

FIG. **3** shows in further detail design optimization step **209** of FIG. **2**.

FIG. **4** shows a parallel processing computer system **400**, suitable for implementing a concurrent design optimizer of the present invention.

FIG. **5**(*a*) shows circuit **500** including circuit elements **501–505** interconnected by nets **512–517**.

FIG. **5**(*b*) shows circuit **520** after placement, routing and logic optimization steps performed under the prior art.

FIG. **5**(*c*) shows circuit **550** after placement, routing and logic optimization steps performed under the present invention.

FIG. **6**(*a*) shows circuit **600** placed and routed in accordance with methods in the prior art.

FIG. **6**(*b*) shows circuit **630** placed, routed and logic optimized in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention provides a performance-driven method for designing an integrated circuit, which can be used in conjunction with a parallel processing computer system. To simplify discussion and to facilitate cross reference among the figures, like elements in the figures are given like reference numerals.

One embodiment of the present invention is a design method **200** illustrated in FIG. **2**. As shown in FIG. **2**, method **200** includes design entry step **101**, functional simulation step **102**, logic synthesis step **104**, floor-planning step **103**, technology checking step **105**, formal verification step **107**, test structure verification step **106**, timing analysis step **108**, gate-level simulation step **109**, front-end processing and pre-layout signoff steps **110**, **111** substantially in the same conventional manner as those steps of like reference numerals illustrated in FIG. **1** and discussed in the previous section. Method **200**, however, synthesizes layout using a novel concurrent design optimization step **209**, which performs in parallel placement, logic optimization and routing functions driven by concurrent timing and power analyses. Design optimization step **209**, discussed in further detail in the following, provides a layout (generated at layout step **210**) which is substantially optimized for power and timing. Accordingly, in method **200**, the number of iterations through the optimization cycle of steps **104–111**, **209** and **210** is substantially less than corresponding optimization cycle of steps **104–113** of method **100** illustrated in FIG. **1**. When the designer is satisfied with the layout design, final timing analysis step **113**, post-layout sign-off step **114** and ATPG step **116** can be performed in the conventional manner.

Concurrent design optimization step **209**, which is illustrated in further detail in FIG. **3**, can be carried out in a parallel processing computer system ("concurrent design optimizer") having an architecture provided, for example, in FIG. **4**.

In FIG. **3**, a gate-level design, which can be provided in the form of an industry standard "netlist", is received at the beginning of concurrent design optimization step **209**. This gate-level design, which results from a logic synthesis step (e.g., logic synthesis step **104**) is then examined to determine if further optimization of the design is required. Typically, the designer indicates whether such optimization is required

when invoking the concurrent design optimizer. If design constraints have not been met, i.e., further optimization is required, an initialization step (i.e., initialization step **301**) is performed.

Initialization step **301** groups the elements of the netlist into clusters. Each cluster is so grouped as to allow highly connected elements to be placed in close proximity. The goal of such clustering is such that circuits within a cluster are tightly coupled and circuits from different clusters are loosely coupled. Proper clustering of the netlist improves performance of the subsequent placement step by reducing the search space.

Upon completing initialization step **301**, a preliminary placement and routing estimation step **302** is performed. Initial placement and routing estimation step **302** (a) maps the clusters into an initial set of partitions ("initial placement"), (b) performs an initial timing and power analysis based on the initial placement, (c) optimizes the initial placement according to a given set of cost functions, and (d) provides an initial routing or an initial set of routing estimates. In addition, a logic optimization step can also be carried out to further improve performance. Suitable logic optimization step includes providing alternative implementations of logic circuits, and insertion or deletion of signal buffers. At this level, as clusters are loosely coupled, significant parallelism can be exploited by a parallel processing computer system, such as computer system **400** of FIG. **4**, which is discussed in further detail below. In one embodiment, the available space in the layout is divided into regions ("bins"), and placement of each cluster is achieved by associating the cluster with a specific bin.

Having provided an initial placement by initial placement and routing or routing estimate step **302**, an iterative process involving an "inner" loop and an "outer" loop is carried out to further refine the design optimization process. Each iteration of the outer-loop reduces the size of each cluster, until the level of individual cells (e.g., logic gates or macrocells) is reached. In addition, each bin can be further partitioned into smaller bins to refine the granularity of the placement.

In the inner-loop, a placement and routing estimate step **303** (a) places the gates in each current cluster into the current set of bins in the physical layout, the current clusters being either the clusters of the initial placement or clusters from the previous outer-loop iteration; (b) providing routing or routing estimates based on the current placement; (c) analyzes timing, routing density and power attributes of this new placement to obtain performance metrics; (d) perform logic optimization on the current placement based on the performance metrics obtained, to obtain an improved gate-level design; and (e) providing an improved set of clusters. As represented by placement and routing or routing estimate step **304**, steps (a)–(d) of placement and routing estimate steps **303** can be repeated to provide further placement optimization, given the current cluster size. Placement optimization step (a) can be carried out using a variety of techniques, such as various annealing techniques, or genetic algorithms. Logic optimization step (c) is performed to synthesize variations of the logic circuits within each cluster, based on which further placement optimization are possible. Step **304** is repeated until some performance constraints are met, e.g., the expected performance gain falls below a threshold metric.

At this point, if the current clusters are not reduced to primitives (e.g., a library cell), a cluster size reduction step **307** divides the current clusters into smaller clusters, and initiates the inner-loop placement and routing or routing estimate steps **303** and **304** in the manner described above, until the clusters become primitives. Then, conventional final timing analysis, post-layout sign-off and ATPG steps **113–116** can be performed.

Because an integrated circuit layout design created from an initial netlist using the method described above, in which placement, routing or routing estimates and logic optimization are performed at every level of cluster refinement, the resulting integrated circuit is expected to be significantly different in organization from an integrated circuit created from the same initial netlist but refined by conventional partitioning, placement and global routing performed in the manner shown in FIG. **1**. For a given gate-level logic netlist, certain optimizations in the layout design attainable using the methods of the present invention are unattainable using conventional serially applied partitioning, placement and global routing methods. FIGS. **5**(*a*)–**5**(*c*) and FIGS. **6**(*a*)–**6**(*b*) provide two examples to illustrate the effects of concurrent placement, routing and logic optimization on integrated circuit designed under the present invention.

As shown in FIG. **5**(*a*), a circuit **500** includes circuit elements **501–505** which are interconnected by nets **512–517**. In circuit **500**, net **518** represents a portion of a "critical" path (i.e., a signal path in which strict timing constraints must be met) which includes circuit elements **505** and **503**. Logic element **501**, however, is not part of the critical path. Thus, many optimization techniques would tend to move circuit elements **505** and **503** closer together to reduce the total wire length of net **518**, hence reducing the interconnect delay.

The prior art technique illustrated by FIG. **1** would provide a circuit implementation such as circuit **530** shown in FIG. **5**(*b*). To reduce the wiring length in net **518**, a placement tool would move circuit elements **501**, **503** and **505** closer together, as shown in FIG. **5**(*b*). Subsequently, a routing tool provides the wiring of nets **512–517**, as shown. However, a subsequent timing analysis would discover that the shorter wire length would still be insufficient to satisfy the strict timing constraints of net **518**. Accordingly, an optimization step after the timing analysis would insert buffer **511** between logic elements **505** and **501**, thereby reducing the capacitance of net **518** and introducing a new net **510** in this non-critical portion of net **518**. However, because logic elements **501** and **503** are moved closer to logic element **501**, the interconnect wiring in nets **512–517** amongst logic elements **501–504** are consequently lengthened, thereby incurring a penalty in circuit area and routing congestion.

Because of the iteration in the inner-loop, however, the present invention would provide an implementation such as circuit **530** shown in FIG. **5**(*c*). During iteration in the inner-loop, the logic optimization step in the inner-loop would recognize hat inserting buffer **511** would satisfy the timing constraints in net **518**, before the placement step in the next iteration in the inner-loop, thus preserving both the initial placement of logic elements **501–504** and the wiring in nets **512–517**.

A second example to illustrate the effects of the present invention on integrated circuit design is provided in FIGS. **6**(*a*) and **6**(*b*). FIG. **6**(*a*) shows a circuit **600**, which includes circuit elements **610–614**. Circuit element **613** includes a decoder which receives encoded signals **620** to **622**, and decodes these signals into signals **623-1** to **623-8**. Signals **623-1** to **623-8** are provided to circuit elements **612** and **614**. Under the prior art placement and routing methods, a

placement tool provides placement of circuit elements **610** to **614**, as shown. Subsequently, a routing tool provides (a) between decoder **613** and **614**, wires **631** for signals **623-1** to **623-8**, (b) between decoder **613** and circuit element **612**, wires **632**a for signals **623-1** to **623-3** in the channel between circuit elements **610** and **611**, and (c) between decoder **613** and circuit element **612**, wires **623**b for signals **623-4** to **623-8** around circuit element **611**. As shown in FIG. **6**(*a*), because placement optimization and routing are only loosely coupled, the placement of circuit elements **610** and **611** allows only three of the eight decoded signals **623-1** to **623-8** to be routed in the narrow channel between circuit elements **610** and **611**. The remaining decoded signals must be routed around circuit element **611**, thus increasing chip area, and causing signal congestion, timing and power dissipation inefficiencies.

In contrast, under the present invention, during iteration in the inner-loop, the placement and logic optimization steps would provide an additional decoder **613'**, as shown in circuit **630** of FIG. **6**(*b*). Signals **623-1** to **623-8** can thus be provided to circuit element **612** from additional decoder **613'**. As shown in FIG. **6**(*b*), relatively short wires **632** and **631** provide signals **623-1** to **623-8** circuit elements **612** and **614**, respectively. Further, the relatively narrow channel between circuit elements **610** and **611** is now used to route encoded signals **620** to **622** to decoders **613** and **613'**. Clearly, if decoder **613'** occupies an area smaller than that occupied by wires **623***b*, circuit **630** of FIG. **6**(*b*) is more efficient in silicon real estate, timing and power dissipation than circuit **600** of FIG. **6**(*a*). Additional decoder **613'** could not have been added under the prior art methods, since there is no room which can be allocated to place an additional decoder.

FIG. **4** shows a parallel processing system **400** suitable for implementing a concurrent design optimizer of the present invention. As shown in FIG. **4**, parallel processing computer system **400** includes a shared memory **420** and a large number of central processing units (CPUs) **410-1**, **410-2**, **410-3**, . . . **410**-(m-**1**) to **410**-m. Provided in shared memory **420** is an operating system **409** which can be executed on each of CPUs **410-1** to **410**-m. A thread library **408** provides the control programs for executing multiple concurrently executed threads on operating system **409**. The concurrent design optimizer provides data structure **401**, representing the design to be optimized, in shared memory **420**. Also residing in shared memory **420** are (a) interface and initialization module **403**, representing the programs for interface with the design database and a user interface, (b) placement module **404**, representing the programs for placement of clusters and for placement optimization; (c) timing/power analysis module **405**, representing programs for incremental timing and power analysis steps, (d) logic optimization module **406**, representing programs for optimizing a logic circuit, and (e) routing or routing estimation module **407**, representing programs for routing and estimating use of routing resources. Alternatively, each CPU can have its own copy of modules **403**–**407**, residing either in shared memory **420** or in a memory to which the CPU has exclusive access (e.g., a private address space).

The algorithms used in the methods described above (e.g., those used in the placement and routing or routing estimate steps **303** and **304** discussed above in conjunction with method **300** of FIG. **3**) can be efficiently executed in parallel. For example, since each of the placement, placement optimization steps, routing and routing estimation, logic optimization, incremental timing and power analysis steps described above operate on at most a few clusters at a time, and since the design includes a large number of groups of clusters, each group involving a small number of clusters can be handled by a different independently executing thread. Each thread can begin execution in any of CPUs **410-1** to **410**-m. Communication within the modules executing within a thread, or between modules executing on different threads can be achieved by a task queue **402**. For example, upon completion of a placement of gates into the lower level clusters, a placement program of placement module **404** running in a thread can create a task in task queue **402**, pointing to a new placement for the thread's assigned clusters in data structure **401**. Upon initialization, a program in timing and power analysis module **405** then accesses task queue **402** to find the task left by the previous placement program and proceeds to perform timing and power analysis on the new placement. To assure data integrity and concurrency under such a highly parallel system, a locking mechanism can be provided to prevent accidental write access by multiple threads to the same data structure (e.g., a cluster). Since granularity of such data structure is small, and the number of such data structures is enormous, the probability of concurrent access by multiple threads to the same data object is small. When multiple threads attempts to access the same data object, a simple resolution mechanism which grants access to one thread and queues accesses by the other threads would suffice.

A CPU, such as any of CPUs **410-1** to **410**-m, would be configured to run any of the programs in modules **403**–**407**, and would be controlled by an operating system which provides services for creating and executing in a thread using programs in thread library **408**. In addition, the CPU would maintain processes which handle reading tasks from and writing tasks to task queue **402**, and reading data from and writing data to data structure **401** which includes the design data of the integrated circuit under design. CPUs **410-1** to **410**-m can be implemented, for example, by a group of high-performance microprocessors, or a group of engineering design workstations.

The above detailed description is provided to illustrate the specific embodiments of the present invention and is not intended to be limiting. Numerous modifications and variations within the scope of the present invention are possible. The present invention is defined by the following claims.

We claim:

1. A method for optimizing a layout design based on a first gate-level design, comprising:

(a) partitioning circuit elements of said first gate-level design into a first plurality of clusters;

(b) mapping each cluster to a portion of said layout design so as to obtain a placement;

(c) based on said placement, providing routing or routing estimates between circuit elements of each cluster and between circuit elements of different clusters;

(d) based on said routing or routing estimates, performing a timing analysis to provide estimates of interconnect delays between circuit elements within each cluster and between circuit elements of different clusters;

(e) performing a logic optimization operation on a selected one of said clusters to obtain a second gate-level logic design based on a cost function, said cost function being a function of said estimates of interconnect delays;

(f) partitioning said second gate-level logic design to obtain a second plurality of clusters; and

(g) repeating (b)–(f) until said cost function is less than a predetermined value.

2. A method as in claim 1, further comprising:

dividing each cluster into a plurality of subclusters; and

applying (b) through (f) on the subclusters.

3. A method as in claim 2, wherein said dividing step and said applying step are recursively applied to said subclusters until said subcluster include only primitive circuit elements.

4. A method as in claim 1, further comprising a placement optimization operation which reassigns circuit elements in a selected cluster to a second cluster mapped to a portion of said layout design adjacent the portion of said layout design mapped to said selected cluster.

5. A method as in claim 1, wherein said logic optimization operation inserts or deletes buffers between circuit elements.

6. A method as in claim 1, wherein said logic optimization operation provides one of several logic circuits performing substantially the same logic function.

7. A method as in claim 1, wherein said plurality of clusters are grouped into a plurality of cluster groups and wherein each of said mapping step, said performing a timing analysis step, said performing a logic optimization operation step and said routing step is applied concurrently to each cluster group using multiple processing units.

8. A method as in claim 1, wherein said clusters are divided so that connectivity amongst elements within a cluster are higher than connectivity amongst elements of different clusters.

9. A method as in claim 1, wherein said circuit elements are provided in a logic net list.

10. A method as in claim 1, further providing an initialization step which includes in said layout design test structures.

11. A method for designing an integrated circuit, comprising:

(a) providing a register-transfer level logic design of said integrated circuit;

(b) performing a functional simulation of said logic circuit;

(c) using a logic synthesis tool, synthesizing a gate-level logic design of said integrated circuit from said register-transfer level logic design of said integrated circuit;

(d) using a timing model, performing a static timing analysis of said gate-level logic design;

(e) placing said gate-level logic design into a first placement representing mappings of circuit elements of said gate-level logic design to physical locations of said integrated circuit, said placing step including interconnect delay-driven circuit placement and logic optimization steps;

(f) extracting from said mappings of circuit elements values of timing parameters of said timing model and updating said timing model;

(g) providing a physical design based on said mappings of circuit elements;

(h) extracting from said physical design and resynthesizing using said logic synthesis tool a second gate-level logic circuit and updating said timing model

(i) repeating (d) to (h) according to a cost function, until such cost function attains value less than a predetermined value; and

(j) performing a final static timing analysis to verify that timing goals are met.

12. A method as in claim 11, wherein (e) comprises:

(a1) partitioning circuit elements of said layout design into a plurality of clusters;

(b1) mapping each cluster to a portion of said layout so as to obtain a placement;

(c1) based on said placement, providing routing or routing estimates for circuit elements within each cluster and between circuit elements of different clusters;

(d1) performing a timing analysis to provide estimates of interconnect delay between circuit elements within each cluster and between circuit elements of different clusters;

(e1) performing a logic optimization operation on a selected one of said clusters to obtain a second gate-level logic design based on a cost function, said cost function being a function of said estimates of interconnect delays;

(f1) based on said second gate-level logic design, providing an improved plurality of clusters; and

(g1) repeating (b1)–(f1) until said cost function is less than a predetermined value.

13. A method as in claim 12, further comprising:

dividing each cluster into a plurality of subclusters; and

applying (b1) through (g1) on the subclusters.

14. A method as in claim 13, wherein said dividing step and applying step are recursively applied to said subclusters until said subcluster include only primitive circuit elements.

15. A method as in claim 12, further comprising a placement optimization operation which reassigns circuit elements in a selected cluster to a second cluster mapped to a portion of said layout design adjacent the portion of said layout design mapped to said selected cluster.

16. A method as in claim 12, wherein said logic optimization operation inserts or deletes buffers between circuit elements.

17. A method as in claim 12, wherein said logic optimization operation provides one of a plurality of logic circuits performing substantially the same logic function.

18. A method as in claim 12, wherein said plurality of clusters are grouped into a plurality of cluster groups and wherein each of said mapping step, said performing a timing analysis step, said performing a logic optimization operation step and routing step is applied concurrently to each cluster group using multiple processing units.

19. A method as in claim 12, wherein said clusters are divided so that connectivity amongst elements within a cluster are higher than connectivity amongst elements of different clusters.

20. A method as in claim 12, wherein said circuit elements are provided in a logic net list.

21. A method as in claim 12, further providing an initialization step which includes in said layout design test structures.

22. A method as in claim 12, wherein said logic optimization operation reassigns circuit elements in a selected cluster to a second cluster mapped to a portion of said layout design adjacent the portion of said layout design mapped to said selected cluster.

23. A parallel processing design automation system for optimizing an integrated circuit design, comprising:

a plurality of central processing units;

a memory for holding said integrated circuit design, said memory being accessible by each central processing units;

a control program managing a task list, said task list specifying tasks for optimizing said integrated circuit design, said control program allocating each of said tasks in said task list to said central processing units said, control program supporting multithread execution;

wherein each said task is one of placement, performance analysis, logic optimization and routing;

wherein said control program provides a locking mechanism for data structures stored in said memory to support concurrent operations by said central processing units each accessing said integrated circuit design represented by said data structures.

24. A parallel processing design automation system as in claim **23**, further configured to include a placement program based upon an algorithm capable of parallel execution by said central processing units accessing said data structures.

25. A parallel processing design automation system as in claim **23**, further configured to include a timing analysis program based upon an algorithm capable of parallel execution by said central processing units accessing said data structures.

26. A parallel processing design automation system as in claim **23**, further configured to include a logic optimization program based upon an algorithm capable of parallel execution by said central processing units accessing said data structures.

27. A parallel processing design automation system as in claim **23**, further configured to include a routing or routing estimation program based upon an algorithm capable of parallel execution by said central processing units accessing said data structures.

28. An integrated circuit having a layout design created by a process comprising:

(a) partitioning circuit elements of a first gate-level logic design into a plurality of clusters;

(b) mapping each cluster to a portion of said layout design so as to obtain a placement;

(c) based on said placement, providing routing or routing estimates between circuit elements of each cluster and between circuit elements of different clusters;

(d) based on said routing or routing estimates, performing a timing analysis to provide estimates of interconnect delays between circuit elements within each cluster and between circuit elements of different clusters;

(e) performing a logic optimization operation on a selected one of said clusters to obtain a second gate-level logic design based on a cost function, said cost function being a function of said estimates of interconnect delays;

(f) partitioning said second gate-level logic design to obtain a second plurality of clusters; and

(g) repeating (b)–(f) until said cost function is less than a predetermined value.

29. An integrated circuit as in claim **28**, said layout design being created by applying further:

dividing each cluster into a plurality of subclusters; and

applying (b) through (f) on the subclusters.

30. An integrated circuit as in claim **29**, wherein said dividing step and applying step are recursively applied to said subclusters until said subcluster include only primitive circuit elements.

31. An integrated circuit as in claim **28**, further comprising a placement optimization operation which reassigns circuit elements in a selected cluster to a second cluster mapped to a portion of said layout design adjacent the portion of said layout design mapped to said selected cluster.

32. An integrated circuit as in claim **28**, wherein said logic optimization operation inserts or deletes buffers between circuit elements.

33. An integrated circuit as in claim **28**, wherein said logic optimization operation provides one of a plurality of logic circuits performing substantially the same logic function.

34. An integrated circuit as in claim **28**, wherein said plurality of clusters are grouped into a plurality of cluster groups and wherein each of said step mapping step, said performing a timing analysis step, said performing a logic optimization operation step and said routing step is applied concurrently to each cluster group using multiple processing units.

35. An integrated circuit as in claim **28**, wherein said clusters are divided so that connectivity amongst elements within a cluster are higher than connectivity amongst elements of different clusters.

36. An integrated circuit as in claim **28**, wherein said logic optimization operation reassigns circuit elements in a selected cluster to a second cluster mapped to a portion of said layout design adjacent the portion of said layout design mapped to said selected cluster.

37. A computer system coupled to a shared memory accessible to a plurality of computer systems, comprising:

a central processing unit;

an operation system which provides services for creating and executing multiple threads;

an interface to said shared memory, said interface reading tasks from and writing tasks to a task queue residing in said shared memory, and reading from and writing to design data of an integrated circuit residing in said shared memory;

a memory containing application programs for processing said design data read from said shared memory and writing back said processed design data to said shared memory through said interface;

wherein said applications programs include programs for performing in a method for optimizing a layout design based on a gate-level design:

(a) partitioning circuit elements of said layout design into a plurality of clusters;

(b) mapping each cluster to a portion of said layout design so as to obtain a placement;

(c) based on said placement, providing routing or routing estimates between circuit elements of each cluster and between circuit elements of different clusters;

(d) based on said routing or routing estimates, performing a timing analysis to provide estimates of interconnect delays between circuit elements within each cluster and between circuit elements of different clusters;

(e) performing a logic optimization operation on a selected one of said clusters to obtain a second gate-level logic design based on a cost function, said cost function being a function of said estimates of interconnect delays;

(f) based on said second gate-level logic design, providing an improved plurality of clusters; and

(g) repeating (b)–(f) until said cost function is less than a predetermined value.

38. A computer system as in claim **37**, said method further comprising:

dividing each cluster into a plurality of subclusters; and

applying (b) through (f) on the subclusters.

39. A computer system as in claim **38**, wherein said dividing step and applying step are recursively applied to said subclusters until said subcluster include only primitive circuit elements.

* * * * *